

Programming for the Web

A Brief History

Joseph Chang - <http://joe.cat> - 04/17/2020

I build information systems
to support online
Sensemaking

**Explore and learn from data to
make informed decisions**

Picking restaurants on Yelp

Literature reviews

Planning a trip

**Explore and learn from data to
make informed decisions**

Researching policies as a voter

Making an important purchase

SearchLens

Expressing and Visualizing User Interests

Lens 1

Lens 2

Lens 3

enabled lenses

disabled lenses

Query Panel

SearchLens

UserID: 4f2cfb41-e136-4515-a85f-004785b965fb

Create

Outdoor seating normal goal importance

balcony

patio

outdoor^{x4}

[click here to enter more keywords](#)

suggestions: [terrace](#) [courtyard](#) [porch](#) [rooftop](#) [fenced](#)

Steak normal goal importance

flank

hanger

sirloin

ribeye^{x4}

steak^{x10}

[click here to enter more keywords](#)

suggestions: [bistec](#) [skirt](#) [hangar](#) [hamburger](#) [swordfish](#)

Alcohol normal goal importance

full

bar^{x2}

byob^{x2}

beer^{x4}

wine^{x4}

[click here to enter more keywords](#)

suggestions: [byo](#) [yuengling](#) [microbrews](#) [drafts](#) [pbns](#)

BYOB (1 keywords)

authentic thai food (3 keywords)

fresh oysters (2 keywords)

Results Panel

Willow Restaurant
★★★★☆ 97 reviews

★★★★★ ...le App. As soon as we pulled up I knew this was the place. The most completely full parking lot to the sweet aromas bellowing out from the kitchen. Once inside we were greeted and told Happy Anniversary by the hostess and taken to our table. A few minutes later our waiter came over to take our drink orders. I went with the Mexican War Streets. It featured Tequila, Aperol, Grand Marnier, Thyme, lime and sugar and the wife had the Gina Marie which featured Pimm's No. 1., strawberry, rhubarb, sugar, lime and house ginger **beer**. My cocktail was delicious and very refreshing an...

★★★★☆ ... restaurant. I've never been on a busy night, so I've always found a comfortable seat either at the **bar** or in the **outdoor** dining area. We went on a warm sunny afternoon. We sat outside. I ordered the kale salad with pecans, goat cheese, and grilled chicken. Dressing was housemaid and delicious. My husband ordered the Beef tartare, he said it was delicious, he also ordered the deviled eggs. The deviled eggs were made with truffle oil, not one of my favorites, but they looked wonderful. His tartare presentation looked nice as well. Our server, I wish I could remember his name, was very knowledge...

[read more...](#)

Gaucha Parrilla Argentina
★★★★☆ 985 reviews

words often mentioned with **ribeye** in the reviews

91 reviews mentioned **ribeye** and 42 of them also mentioned **sandwich**

★★★★★ ...ch along the walls. More seating outside. Wood everywhere! FOOD We ordered the top two recommended sandwiches: Ribeye sandwich (medium rare, unless you tell them otherwise) and the Rosemary braised b...

★★★★★ Awesome **ribeye** steak sandwich, awesome corn, the staff is very friendly, the food speaks for itself with the flavor of the wood in everything. The only thing I didn't care for was the soggy bun, it di...

★★★★☆ ...w on to the "warm fuzzies": The sandwich I had was unbelievably good! Quite possibly the very best sandwich in the Burg. I got the Carne with **ribeye** steak. It was a generous serving of real ribeye st...

★★★★★ ...ag my boyfriend here but he was thanking me afterwards! Skip the filet mignon cut and get the steak sandwich with **ribeye** cut. **Ribeye** just tastes better than filet mignon on bread. It comes with roaste...

★★★★☆ ...tested! But nevertheless, I did not enjoy the charred, ashy flavor of their meats (particularly the **ribeye** steak sandwich). I didn't add any additional toppings (which might have helped), but I wanted...

★★★★★ ...ed inside at all. We asked one of the staff what the most popular items were, and he said the carne sandwiches were the hot item. I went with a **ribeye** carne sandwich (you can choose from different cut...

★★★★★ ... for the outdoors spot since it was a nice day, and it was great. My friend and I ordered the carne sandwich (with filet, medium rare), a **ribeye** plate, empanadas, and grilled corn. We were hungry. C...

Stars ██████████

Outdoor seating ██████████

Steak ██████████

Alcohol ██████████

Overview Panel

Meta

Compare products across sources

The Best Carry-on Luggage, According to Travel Editors

Hardside or softside, two-wheel or spinner, suitcase or backpack: whatever your preference, we've selected the top carry-on bags in every category.

BY TRAVEL + LEISURE STAFF | JANUARY 04, 2020

Each product we feature has been independently selected and reviewed by our editorial team. If you make a purchase using the links included, we may earn commission.

PHOTO: GETTY IMAGES

We know the original "Keep Calm and Carry On" slogan wasn't initially intended to have anything to do with luggage, but we really do believe in skipping a checked bag whenever possible. With all of your belongings on board, you can

Summary

Brands by Source

	TL	UP	TL	UP	TL
Travelpro	✓	✓	✓	✓	✓
Briggs & Riley	✓	✓	✓	✓	
Samsonite		✓	✓	✓	✓
DELSEY Paris		✓	✓	✓	
AmazonBasics		✓	✓	✓	
Wrangler	✓			✓	
Victorinox		✓	✓		
Hook + Albert		✓	✓		
Rimowa		✓	✓		
Osprey		✓		✓	
Kenneth Cole REACTION			✓	✓	
CHESTER			✓	✓	
Timbuk2			✓		✓
Relavel		✓			
Enkloze		✓			
TUMI		✓			
eBags		✓			
Mancro				✓	
Decolighting				✓	
Reliable Hardware Company				✓	
Kah&Kee				✓	
SwissGear				✓	
Cabin Max				✓	
Hynes Eagle				✓	
NOMATIC			✓		
Solgaard			✓		
Bric's			✓		
JetKids			✓		

Sources

All Options (74)

My Options (0)

Programming of the Web

A Brief History

Joseph Chang - <http://joe.cat> - 04/17/2020

Agenda

What are the major paradigm shifts?

What pushed the community towards these paradigm shifts?

Upcoming trends...

(Modern Web development and the React Framework)

Paradigm Shifts

- Static Webpages -> Dynamic Webpages
- Dynamic HTML and Ajax (so called "Web 2.0")
- Imperative -> Declarative

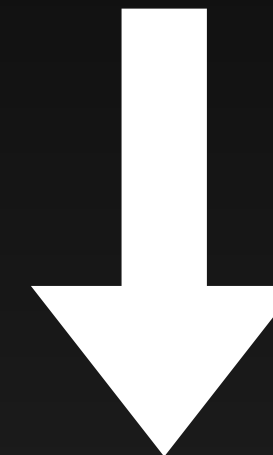
“The dream behind the Web is of a common information space in which we communicate by sharing information.

**Its universality is essential:
A hypertext link can point to anything”**

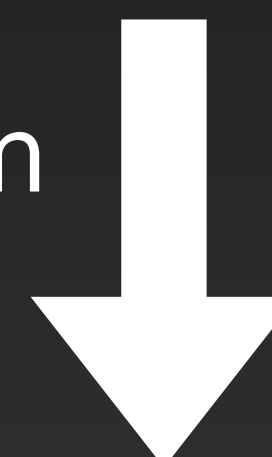
Tim Berners-Lee, 1989

Static Webpages

<http://www.cs.cmu.edu/~bam/uicourse/830spring20/schedule.htm>



~bam/uicourse/830spring20/schedule.htm



schedule.htm



www.cs.cmu.edu

schedule.htm

```

<h1>
  Advanced User Interface Software
</h1>
<h5>
  05-830 Spring 2020
</h5>

<h3>
  Course Description
</h3>
<p>
  This course will cover the research and implementation of user interface
  software. The students will get a comprehensive understanding of all the
  approaches that have been investigated by researchers and commercial systems
  for user interface software. This will be of value to people planning to be
  user interface researchers or implementers, or people interested in learning
  how to provide user interface frameworks for others.
</p>

<table>
  <tbody><tr>
    <td>Time and Place:</td>
    <td>
      1:30PM - 2:50PM in GHC 4301
    </td>
  </tr>
  <tr>
    <td>Instructor:</td>
    <td>
      <a href="https://www.cs.cmu.edu/~bam/">
        Brad A. Myers
      </a>
      <a href="mailto:bam@cs.cmu.edu">
        bam@cs.cmu.edu
      </a>
    </td>
  </tr>
  <tr>
    <td>Office:</td>
    <td>NSH 3517</td>
  </tr>
</tbody>
</table>

```

Advanced User Interface Software

05-830 Spring 2020

Course Description

This course will cover the research and implementation of user interface software. The students will get a comprehensive understanding of all the approaches that have been investigated by researchers and commercial systems for user interface software. This will be of value to people planning to be user interface researchers or implementers, or people interested in learning how to provide user interface frameworks for others.

Time and Place: 1:30PM - 2:50PM in GHC 4301

Instructor: [Brad A. Myers bam@cs.cmu.edu](https://www.cs.cmu.edu/~bam/)

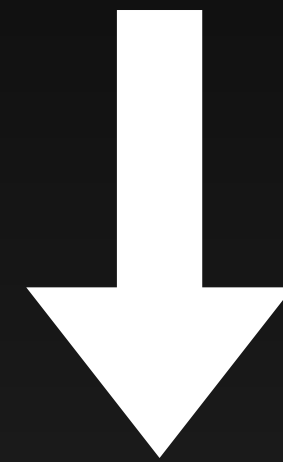
Office: NSH 3517


```
<tr>
  <td>Instructor:</td>
  <td>
    <a href="https://www.cs.cmu.edu/~bam/">
      Brad A. Myers
    </a>
    <a href="mailto:bam@cs.cmu.edu">
      bam@cs.cmu.edu
    </a>
  </td>
</tr>
```

Instructor: [Brad A. Myers](https://www.cs.cmu.edu/~bam/) bam@cs.cmu.edu

Static Webpages

www.cs.cmu.edu/~bam/index.htm



~bam/uicourse/index.htm



index.htm



www.cs.cmu.edu



School of Computer Science

Carnegie Mellon University, Pittsburgh PA 15213-3891
scs@cs.cmu.edu. (412)268-8525 . (412)268-5576 (fax)

This is the main entrance to the School of Computer Science for people coming in from outside the School. If you're looking for information more relevant to SCS residents, try the [SCS Insider's Page](#).

Departments and Institutes

[Computer Science](#), [Robotics](#), [Human-Computer Interaction](#), [Language Technologies](#), ...

Education

[B.S. in C.S.](#), [Masters](#), [Doctoral](#), [Distance Education](#), [Courses & Seminars](#)

People

[Search](#), [Browse](#), [Who's Who](#), [Faculty Lists](#)

Research

Publications

[Technical reports](#), [Videotape Collections](#), [Theses](#), [Digital Library](#), [CMU Libraries](#), ...

Information Please

[About SCS](#), [SCS Career Center](#), [Directions](#)


News and Events

[Alumni news](#), [SCS Today](#), [Calendar](#), [News Releases](#), ...

See the [SCS Insider's Page](#) for info relevant to SCS residents.
The [School of Computer Science](#) is part of [Carnegie Mellon University](#).
This page maintained by webmaster@cs.cmu.edu.

Game 1 - Addventure Classic x +

Not Secure | addventure.com/addventure/game1/



Note: This game, unlike [other games](#) in the Addventure series, is *not expandable*. You will receive an error message when you get to a room that was never written.

Addventure Classic was first set running on 1 June 1994 and ran until 19 Mar 1995. Over 10,000 rooms were created, with a total of 30,000 possible at the time it was terminated.

We are pleased to bring it to you for your reading... er... pleasure. At this time, the game will terminate rather ungracefully when you enter an uncreated room, and we hope to remedy this in the future. [Let us know](#) if you find any problems that seem abnormal.

[Read](#) the story.

Please send us your [comments](#) about this game, especially if you find any problems.

[Whats new](#) with this story or page. (Last updated 27 Aug 1996)

Go to the [Addventure](#) home page.

Addventure room 1

Ready... set...

[Play!](#)

Addventure room 2

The place is deep, dark, and mysterious. This is wrong. All wrong. It wasn't supposed to be like this.

But you have no choice now, do you? No choice but the staircase in front of you.

[Go up](#)

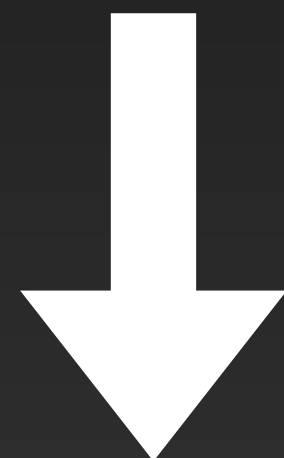
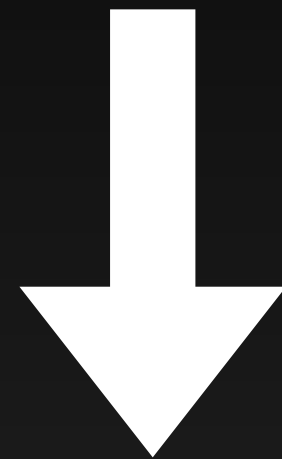
[Go down](#)

Paradigm Shifts

- Static Webpages -> Dynamic Webpages
- Dynamic HTML and Ajax (so called "Web 2.0")
- Imperative -> Declarative

Dynamic Webpages

<http://www.cs.cmu.edu/program.cgi>



program.cgi



<html generated on-the-fly>



Execute program.cgi

www.cs.cmu.edu


```
<form action="http://cs.cmu.edu/action_page.cgi">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname"><br>  
  <input type="submit" value="Submit">  
</form>
```

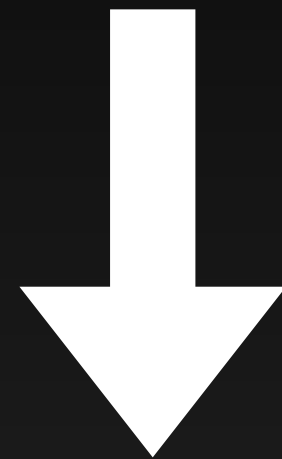
First name:

Last name:

Submit

Dynamic Webpages

<http://www.cs.cmu.edu/program.cgi>



program.cgi
<parameters>



<generated html>



Execute program.cgi

www.cs.cmu.edu

First name:

Last name:



Hello, Joseph Chang

Perl

```
#!/usr/bin/perl
use strict;
use CGI qw(:standard);

my $name= param('firstname');
print header;
print html(
    body(
        p("Hello, $name!"),
    ),
);
```

PHP

```
<html>
<body>
<h1>User Greeting</h1>
<?php
    // This will print whatever the user entered into the form.html page.
    $name = filter_input(INPUT_GET, 'firstname',
    FILTER_SANITIZE_STRING);
    echo "Hello, ". $name . "!";
?>
</body>
</html>
```

PHP

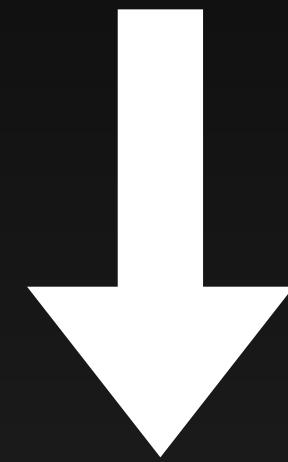
```
<html>
<body>
<h1>User Greeting</h1>
<?php
    // This will print whatever the user entered into the form.html page.
    $name = filter_input(INPUT_GET, 'firstname',
FILTER_SANITIZE_STRING);
    echo "Hello, ". $name . "!";
?>
</body>
</html>
```

<?php
...
?>

Program logic embedded in HTML

Dynamic Webpages

<http://www.cs.cmu.edu/program.cgi>



program.cgi
<parameters>



www.cs.cmu.edu

<generated html>



Execute program.cgi



Database

Stateful!

Welcome to my web site.
You are visitor number

021529



Navigation: About Downloads Customise Support Development Blog **Community** Hosting

Quick links: Quick links FAQ Register Login

Board index

It is currently Tue Mar 05, 2019 8:09 pm

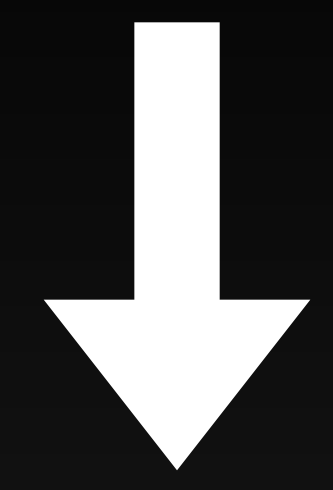
GENERAL	TOPICS	POSTS	LAST POST
 Announcements Read me first before posting anywhere! Subscribe to the feed, available in Atom or RSS format.	428	643	phpBB 3.2.5 Release by Marc Sat Dec 22, 2018 4:35 pm
 Area 51 Development Board Visit the Area 51 development board, phpBB's testing ground of bleeding edge developmental code, to discuss development & code changes, RFCs, future versions of phpBB, and also take a peek at the currently available development version of phpBB, if one is available.	Total redirects: 412199		

SUPPORT	TOPICS	POSTS	LAST POST
[3.2.x] Support Forum Get help with installation and running phpBB 3.2.x here. Please do not post bug reports, feature requests, or extension related questions here. Subforums: [3.2.x] Convertors , [3.2.x] Translations	7569	47368	Re: Captcha errors by lochness Tue Mar 05, 2019 6:00 pm

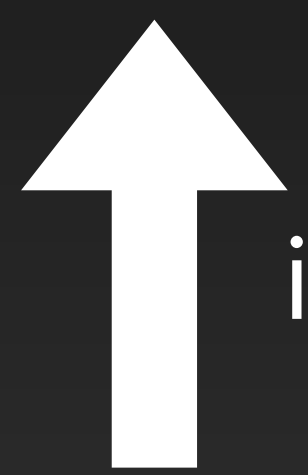
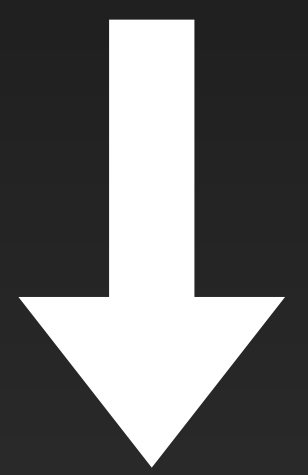
Paradigm Shifts

- Static Webpages -> Dynamic Webpages
- [Dynamic HTML and Ajax](#)
- Imperative -> Declarative

<http://www.cs.cmu.edu/~bam/index.htm>



~bam/index.htm



index.htm



www.cs.cmu.edu

Every User Interaction

=

Full page load

First name:

Last name:

Submit



School of Computer Science
Carnegie Mellon University, Pittsburgh PA 15213-3891
scs@cs.cmu.edu. (412)268-8525 . (412)268-5576 (fax)

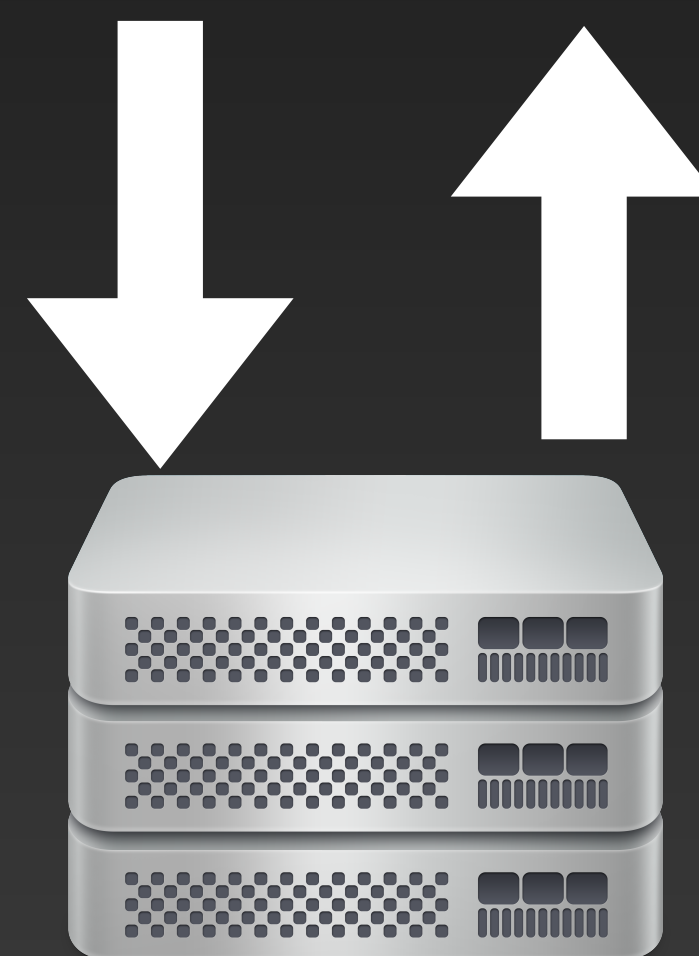
This is the main entrance to the School of Computer Science for people coming from outside the school. For more information more relevant to SCS residents, try the [SCS Insider's Page](#).

Departments and Institutes
[Computer Science](#), [Robotics](#), [Human-Computer Interaction](#), [Language](#)

Education
[B.S. in C.S.](#), [Masters](#), [Doctoral](#), [Distance Education](#), [Courses & Sem](#)

Dynamic HTML / "dHTML" (1997-)

<http://www.cs.cmu.edu/>



<html> + <javascript>

www.cs.cmu.edu

Dynamic HTML (1997-)



The image shows a side-by-side comparison of a code editor and a web browser. The code editor on the left, titled 'Default (vim)', contains the following HTML and JavaScript code:

```
<h1>Click Me</h1>
<script>
  var element = document.getElementsByTagName('h1')[0]
  element.onclick = function() {
    if (element.style.color !== 'green') {
      element.style.color = 'green'
    } else {
      element.style.color = 'red'
    }
  }
</script>
```

The browser on the right, titled 'demo2.html', displays the rendered page. The text 'Click Me' is shown in a large, bold, black font. A mouse cursor is positioned over the text, indicating it is interactive. The browser's address bar shows the file path '/Users/josepl'.

Dynamic HTML (1997-)

```
Default (vim) 381
<h1>Click Me</h1>
<div>container</div>

<script>
  var element = document.getElementsByTagName('h1')[0]
  var container = document.getElementsByTagName('div')[0]

  element.onclick = function() {

    var newNode = document.createElement("div")
    newNode.textContent = 'New Element!'
    container.appendChild(newNode)


  }
</script>
```

Click Me

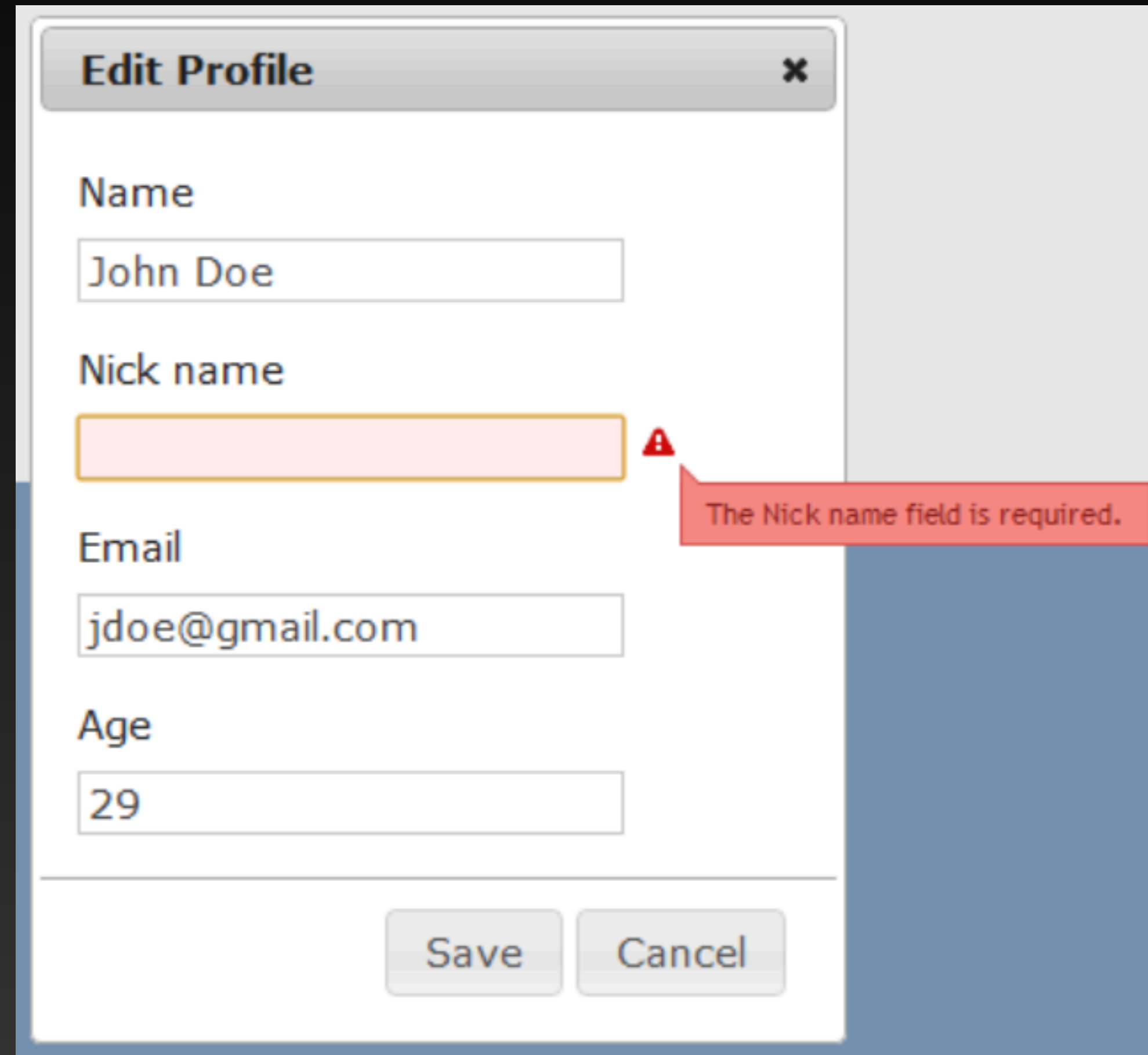
container

Selecting and Manipulating HTML Elements

```
<h1>Click Me</h1>
<script>
  var element = document.getElementsByTagName('h1')[0]
  element.onclick = function() {
    if (element.style.color !== 'green') {
      element.style.color = 'green'
    } else {
      element.style.color = 'red'
    }
  }
</script>
```



Client-side Form Validation



The image shows a web form titled "Edit Profile" with a close button (x) in the top right corner. The form contains four input fields: "Name" (containing "John Doe"), "Nick name" (empty and highlighted in red), "Email" (containing "jdoe@gmail.com"), and "Age" (containing "29"). At the bottom of the form are "Save" and "Cancel" buttons. A red error message box is positioned to the right of the "Nick name" field, containing the text "The Nick name field is required." and a small red warning icon.

Edit Profile x

Name
John Doe

Nick name

Email
jdoe@gmail.com

Age
29

Save Cancel

The Nick name field is required.

Dynamic Webpages - JS Games

Elizium - Dark Rock Music

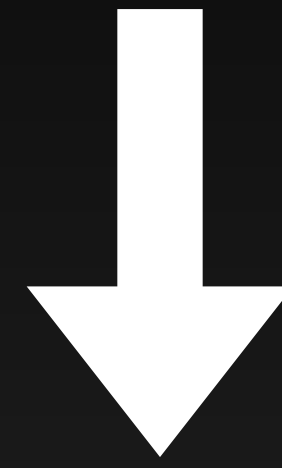


```
<!doctype html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>...</head>
<frameset rows="1,*" border="0" framespacing="0">
  <frame name="bgmusic" src="html/playmusic.php?file=lemmings.mid" frameborder="0" scrolling="no" noresize="noresize" marginwidth="0" marginheight="0">...</frame>
  <frame name="main" src="html/preload.php" frameborder="0" scrolling="yes" noresize="noresize" marginwidth="0" marginheight="0">
    <#document
      <!-- DHTML Lemmings(TM) - GNU Copyright (c) 2004 by crisp / freesoftware[at]xs4all.nl -->
      <!doctype html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
      <html xmlns="http://www.w3.org/1999/xhtml">
        <head>...</head>
        <body onload="preload_images()" marginwidth="0" marginheight="0" style="margin-bottom: 47px;">
          <center>...</center>
          <div id="content" style="overflow: hidden">
            <div id="loading" style="visibility: hidden;">...</div>
            <div id="progressbar" style="width: 560px; visibility: hidden;">&nbsp;&nbsp;&nbsp;</div>
            <div id="playground" style="width: 1030px; left: -281px; visibility: visible;">
              
              
              
              
              <div class="lemming" style="top: 118px; left: 374px; background-image: url("../img/lemming_walk_r.gif"); background-position: 0px center;"></div>
              <div class="lemming" style="top: 112px; left: 298px; background-image: url("../img/lemming_walk_l.gif"); background-position: -128px center;"></div>
              <div class="lemming" style="top: 116px; left: 386px; background-image: url("../img/lemming_walk_l.gif"); background-position: 0px center;"></div>
              <div class="lemming" style="top: 118px; left: 474px; background-image: url("../img/lemming_walk_l.gif"); background-position: -128px center;"></div>
              <div class="lemming" style="top: 126px; left: 564px; background-image: url("../img/lemming_walk_l.gif"); background-position: -224px center;"></div>
              <div class="lemming" style="top: 116px; left: 636px; background-image: url("../img/lemming_walk_r.gif"); background-position: -96px center;"></div>
              <div class="lemming" style="top: 128px; left: 542px; background-image: url("../img/lemming_walk_r.gif"); background-position: -128px center;"></div>
              <div class="lemming" style="top: 112px; left: 450px; background-image: url("../img/lemming_walk_r.gif"); background-position: -192px center;"></div>
              <div class="lemming" style="top: 116px; left: 360px; background-image: url("../img/lemming_walk_r.gif"); background-position: -32px center;"></div>
              <div class="lemming" style="top: 110px; left: 316px; background-image: url("../img/lemming_walk_l.gif"); background-position: -96px center;"></div>
            </div>
            <div id="scrollbar" style="visibility: visible;">...</div>
          </div>
          <div id="controls" style="visibility: visible;">...</div>
          <div id="statusbar" style="visibility: visible;">...</div>
        </body>
      </html>
    </frame>
  </frameset>
</html>
```

Save 1 of 10 Lemmings Out: 10 Home:

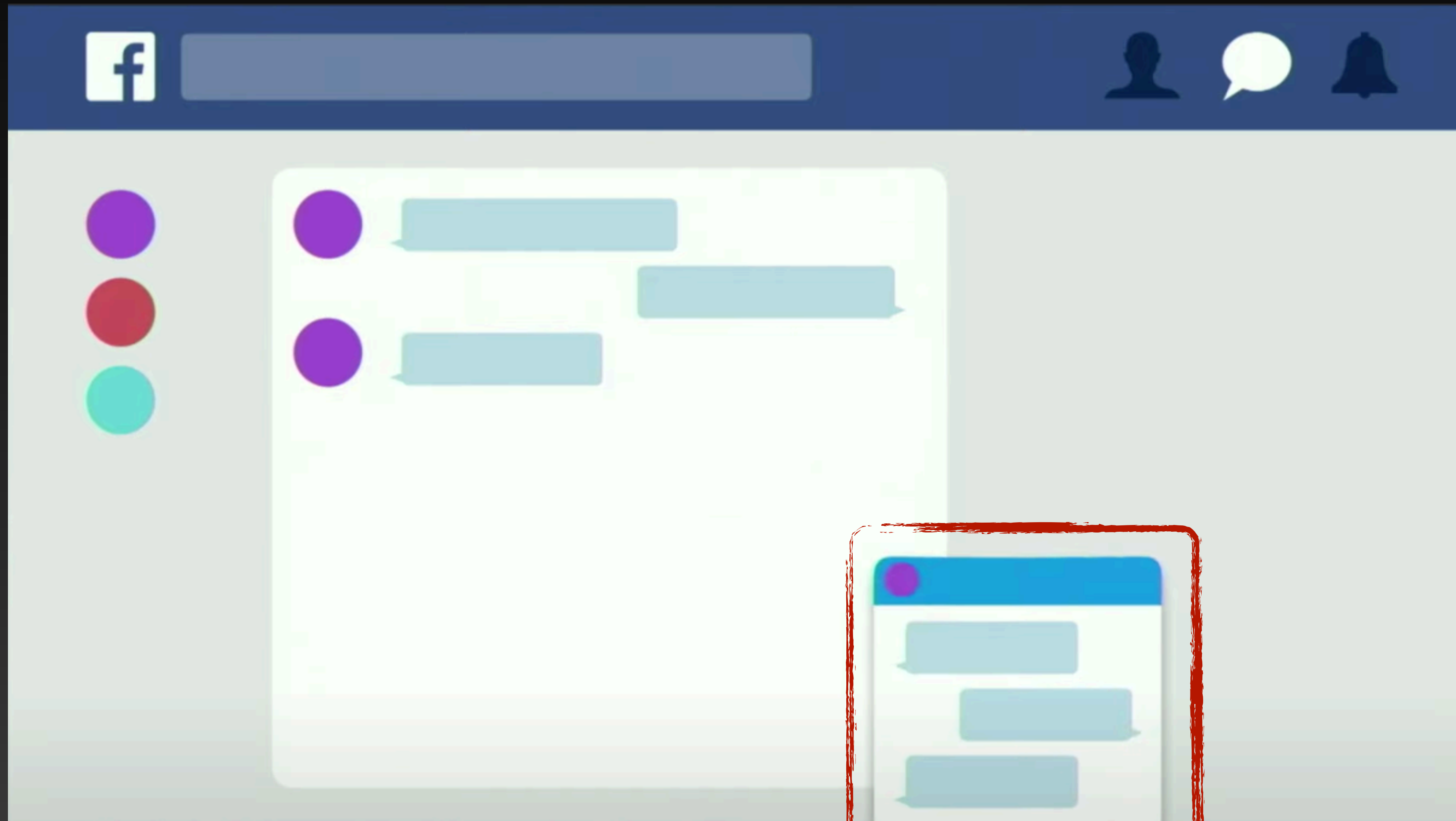
dHTML (1997-)

<http://www.cs.cmu.edu/>



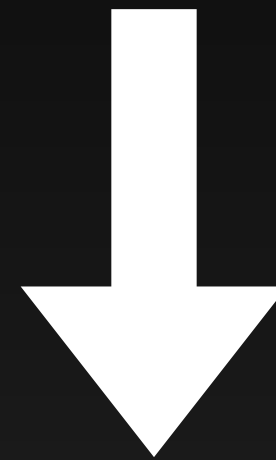
www.cs.cmu.edu

Embedded Online Chat



Ajax (1999-)

<http://www.cs.cmu.edu/>

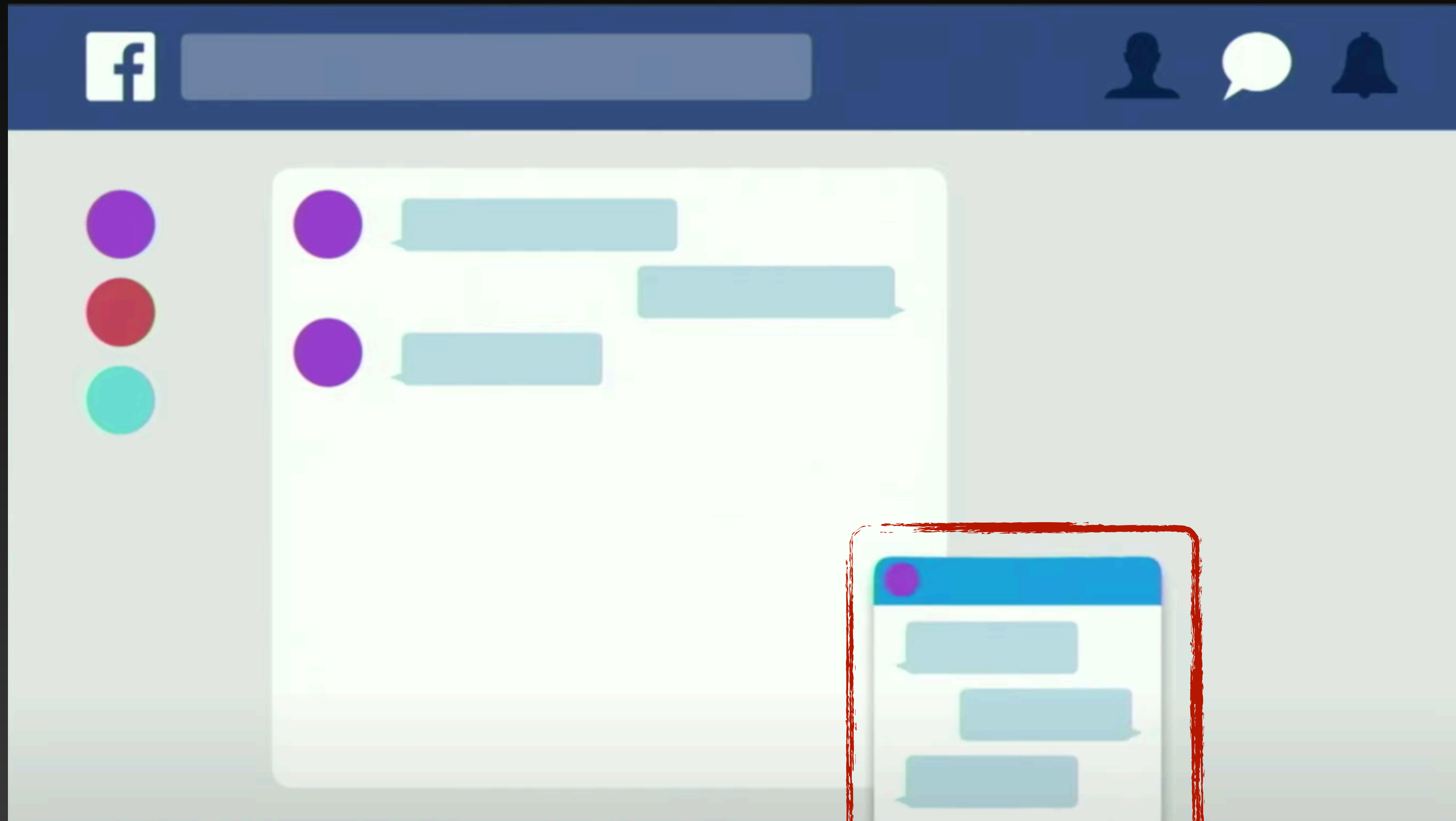


www.cs.cmu.edu



additional requests
without page load

Embedded Online Chat



Shared Documents

The screenshot shows a Google Docs interface for a document titled "process". The document content is as follows:

When not actively doing a lit review

- Come across interesting articles from google scholar / twitter / friends
- Sometimes read immediately sometimes put it aside
- Rarely take notes
 - but would like to if I know where to put it / how to find it when I need to?
- If it's really relevant, save it somewhere in my OneNote / Fuse and write a note about it
 - Relevant usually means that
 - I already have a structure for it somewhere
 - "I can probably cite this for what I'm currently doing"

When actively conducting a review

Finding sources to read

- Start with a couple of seed papers that I know are relevant + interesting
 - Usually something I've already read in the past
 - Look at citations (bi-directional) and add potential sources to a queue
 - Use semantic scholar to look at citation context
- Start with some keywords
 - Usually pretty vague and unsure what the common terms were
 - "Visualization explanation personalization search results"
 - "Interactive visual explanation for machine learning"
 - Iterate query terms while adding potential sources in search results to a queue (usually tabs first and then OneNote)
- Go through proceedings of recent conferences
- Try to find related workshops / shared tasks

A comment from Joseph Chee is visible on the right side of the document, dated 5:34 PM Today, with the text "blah blah".

Web Pages -> Web Applications

Backend -> Frontend



- Cross-browser compatibility
- Made it easier to select and manipulate elements in the DOM tree
- Impact:
 - Developers started building complex and reusable components

Selecting Elements with jQuery

jQuery("#container > .button")

```
var container =  
    document.getElementById("container")  
var buttons = []  
for (var child in container.children) {  
    if (child.className == "button") {  
        buttons.push(child)  
    }  
}
```

```
<div id='container'>  
    <div>....</div>  
    <div class='button'>....</div>  
    <div class='button'>....</div>  
    <div>....</div>  
</div>
```


Selecting Elements with jQuery

`jQuery("#container .button")`

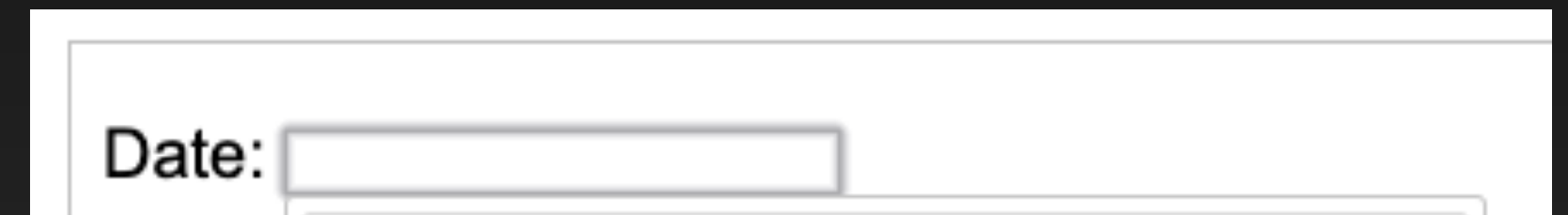
DOM tree traversal

```
<div id='container'>  
  <div>...</div>  
  <div class='button'>...</div>  
  <div>  
    <div class='button'>...</div>  
  </div>  
  <div>...</div>  
</div>
```

jQuery Plugins

Reusable components with both program logic and DOM structures

```
<input id="picker" value="...">
```



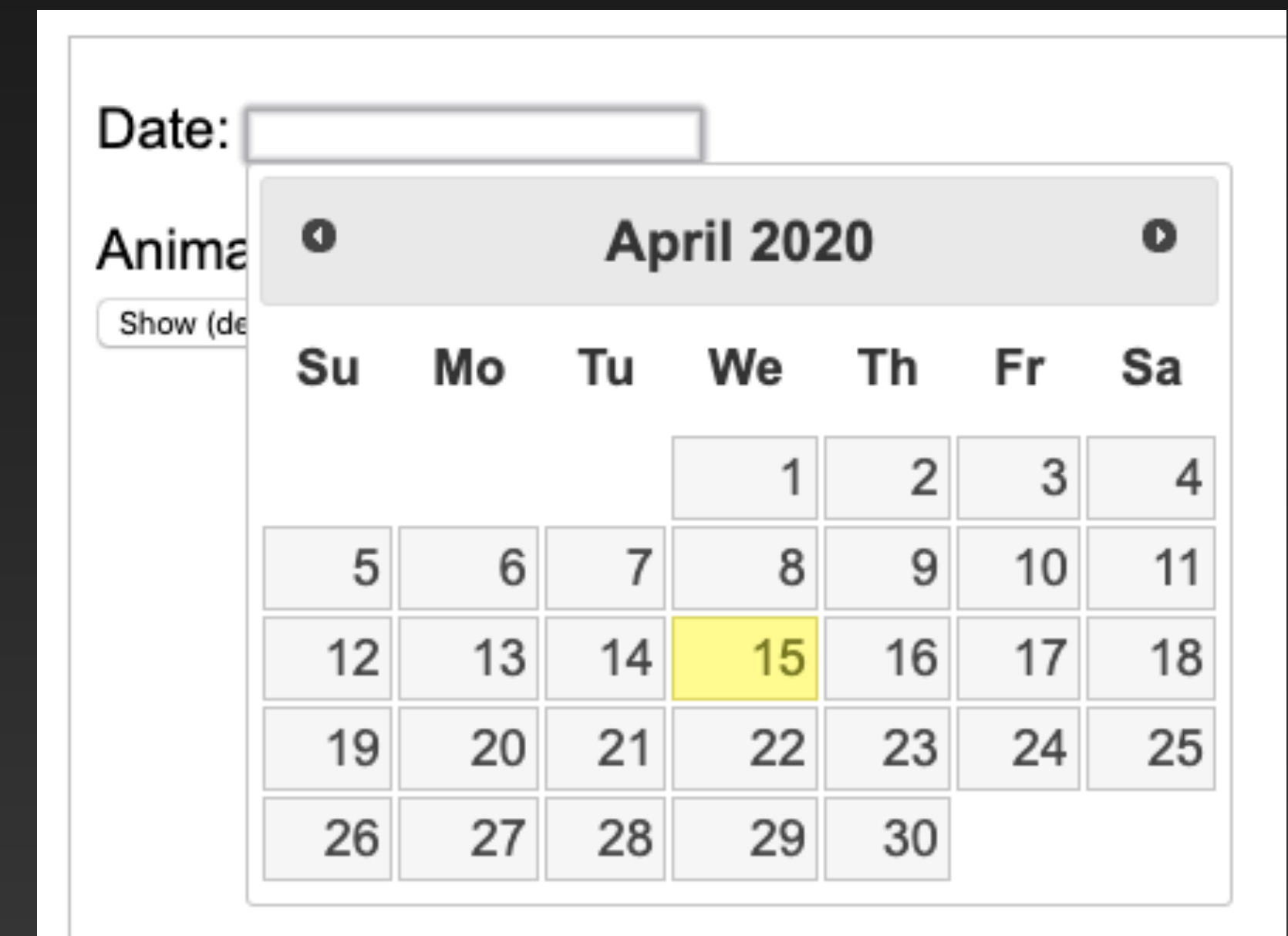
```
jQuery( "#picker" ).datepicker()
```

jQuery Plugins

Reusable components with both program logic and DOM structures

```
<input id="picker" value="...">
```

```
jQuery( "#picker" ).datepicker()
```



jQuery Plugins



jquery




[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)



Repositories	207K
Code	262M+
Commits	2M
Issues	700K
Discussions Beta	1
Packages	652
Marketplace	0
Topics	532
Wikis	74K
Users	6K

Languages	
JavaScript	91,702
HTML	52,011
CSS	18,283
PHP	8,901
Java	2,624



jQuery




jQuery is a lightweight library that simplifies programming with JavaScript.

[See topic](#)

[★ Star](#)

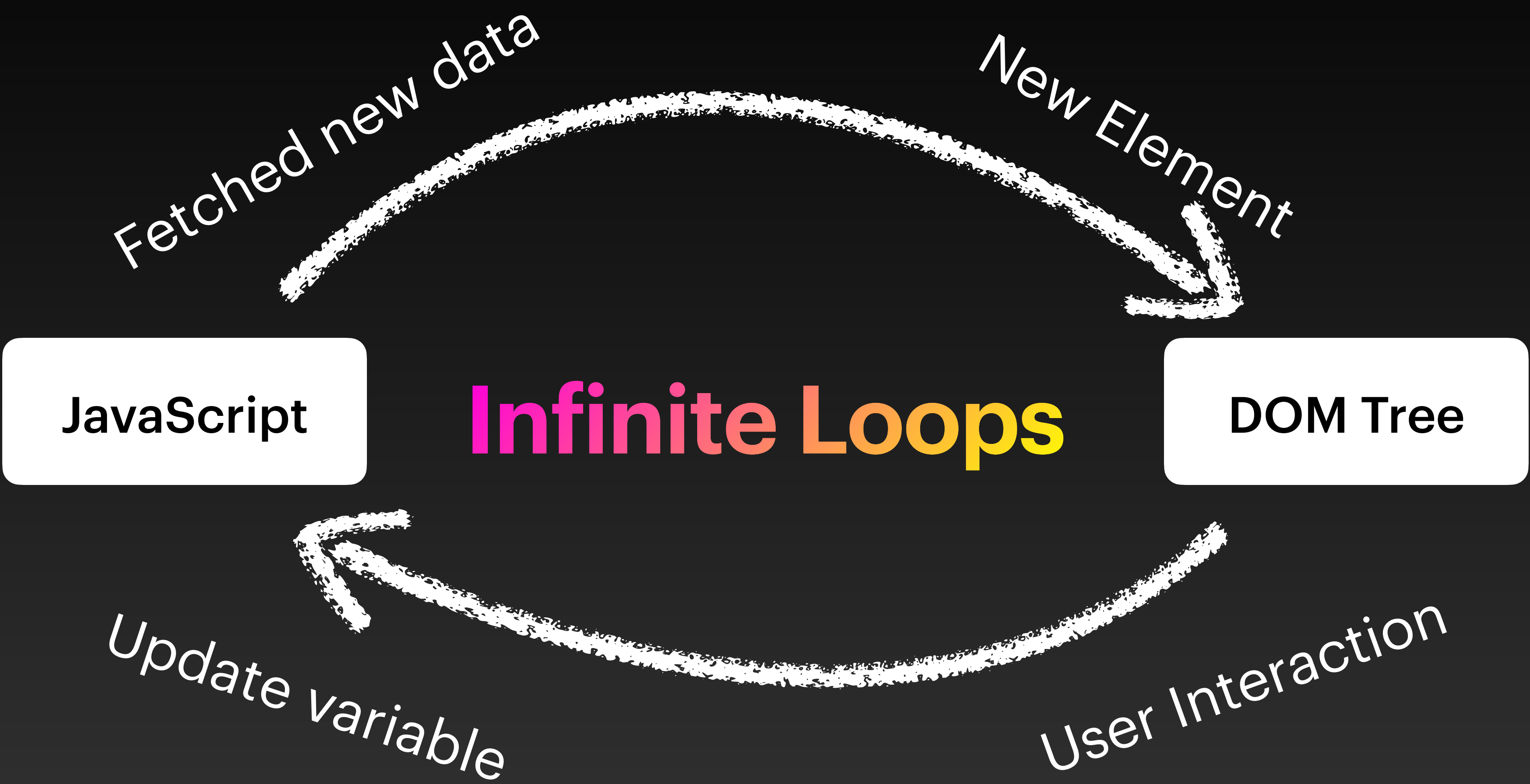
207,937 repository results

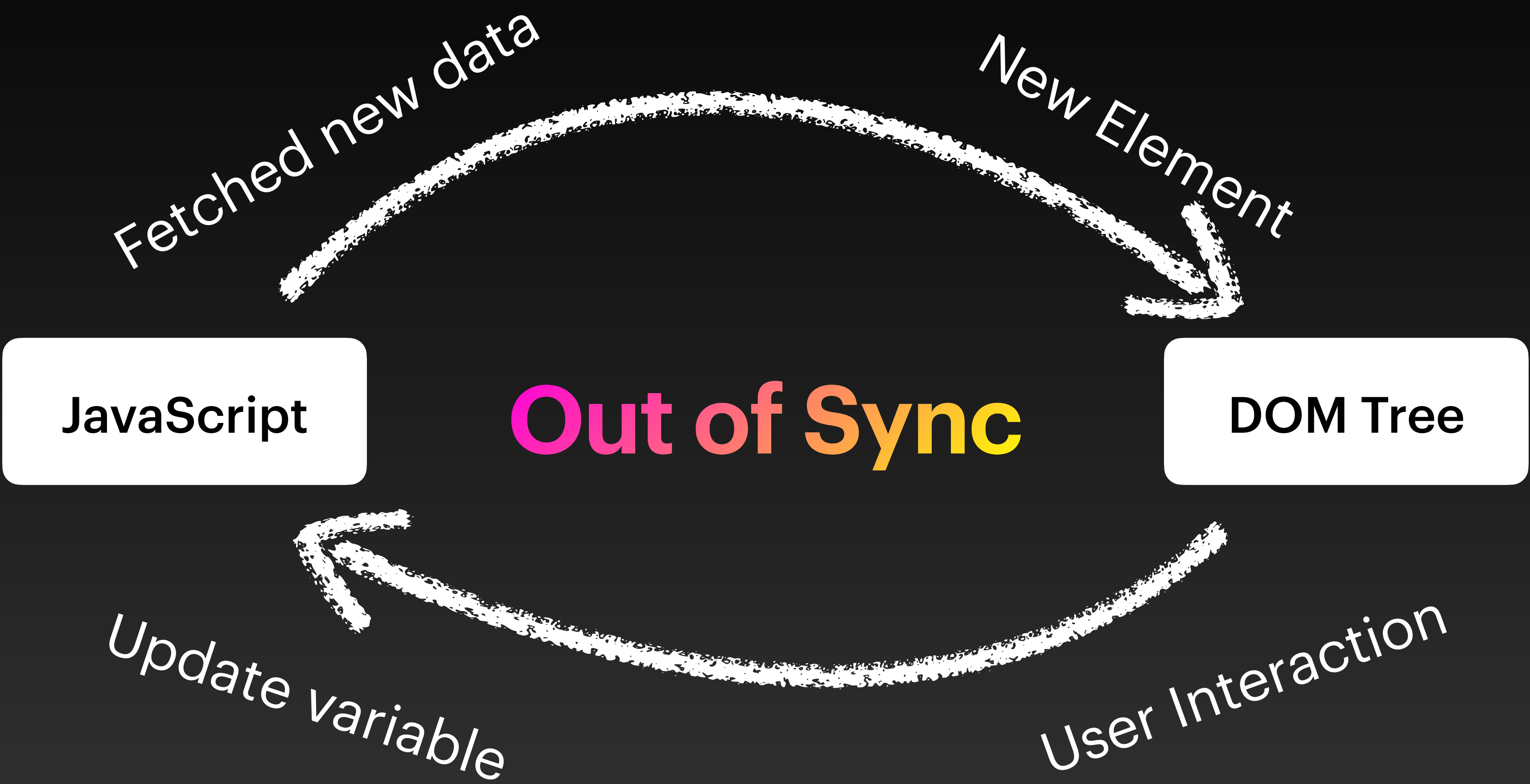
Sort: **Best match** ▾

-  [loopj/jquery-tokeninput](#) Archived
Tokeninput is a **jQuery** plugin which allows your users to select multiple items from a predefined list, using autocomp...
★ 2.4k ● JavaScript Updated on May 31, 2018
-  [kamens/jQuery-menu-aim](#)
jQuery plugin to fire events when user's cursor aims at particular dropdown menu items. For making responsive mega dr...
★ 7.7k ● JavaScript Updated on Nov 1, 2018
-  [jquerytools/jquerytools](#)
The missing UI library for the Web
★ 1.8k ● JavaScript Updated on Sep 18, 2017

jQuery Pitfalls

- DOM tree with thousands+ of nested elements can be tedious to maintain and manipulation programmatically
- Manually maintaining and syncing two sets of states
 - DOM element attributes
 - Javascript variables





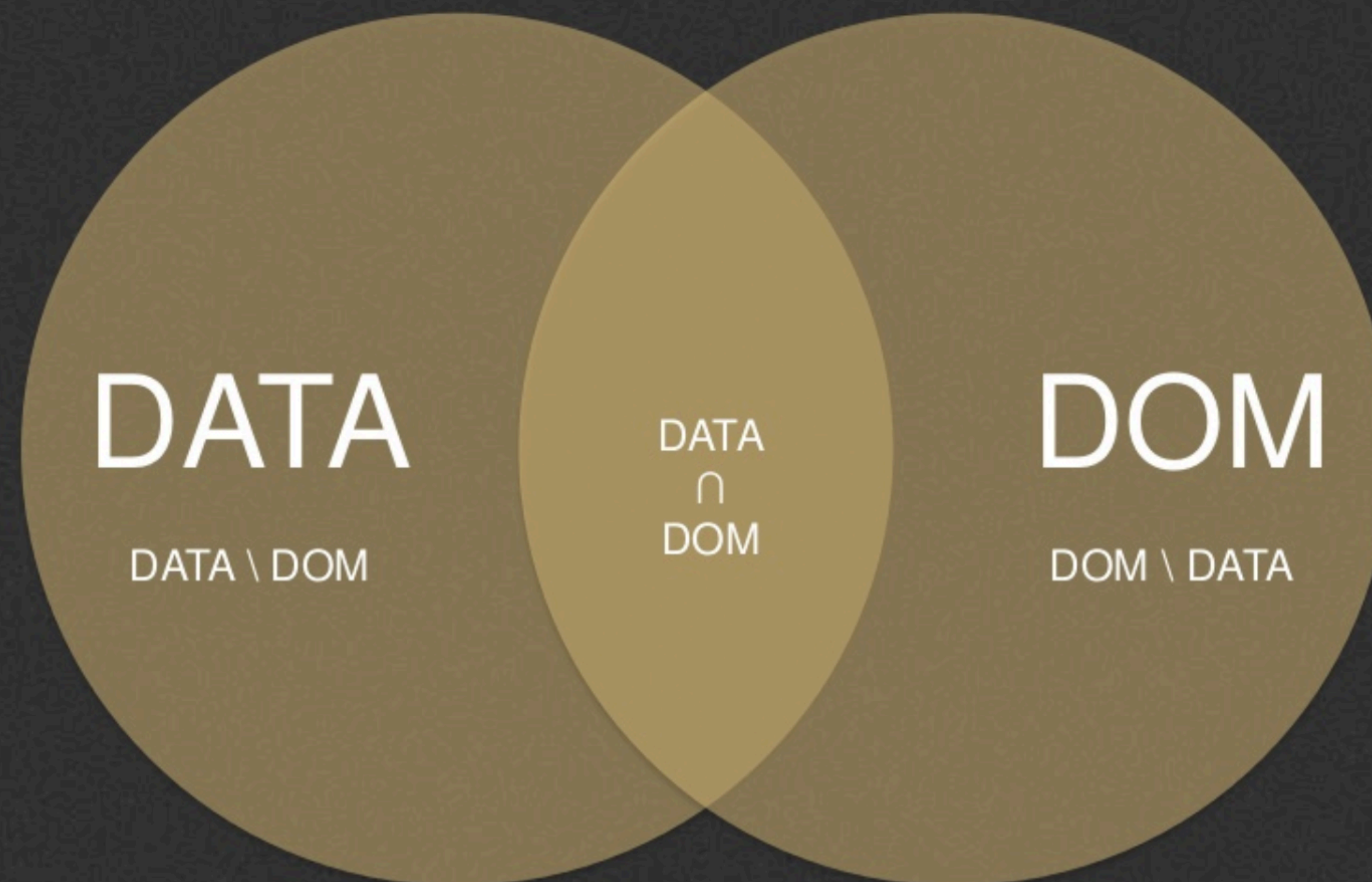
Paradigm Shifts

- Static Webpages -> Dynamic Webpages
- Dynamic HTML and Ajax (so called "Web 2.0")
- Imperative -> Declarative

D3 (2011) - Data-driven Documents

The Enter, Update, Exit Pattern

Typical pattern for binding data to the DOM





.JSX

- Declarative
 - No DOM manipulation
- Component-based
 - Reuse and encapsulation
 - Compose different components
 - One-way Data Flow

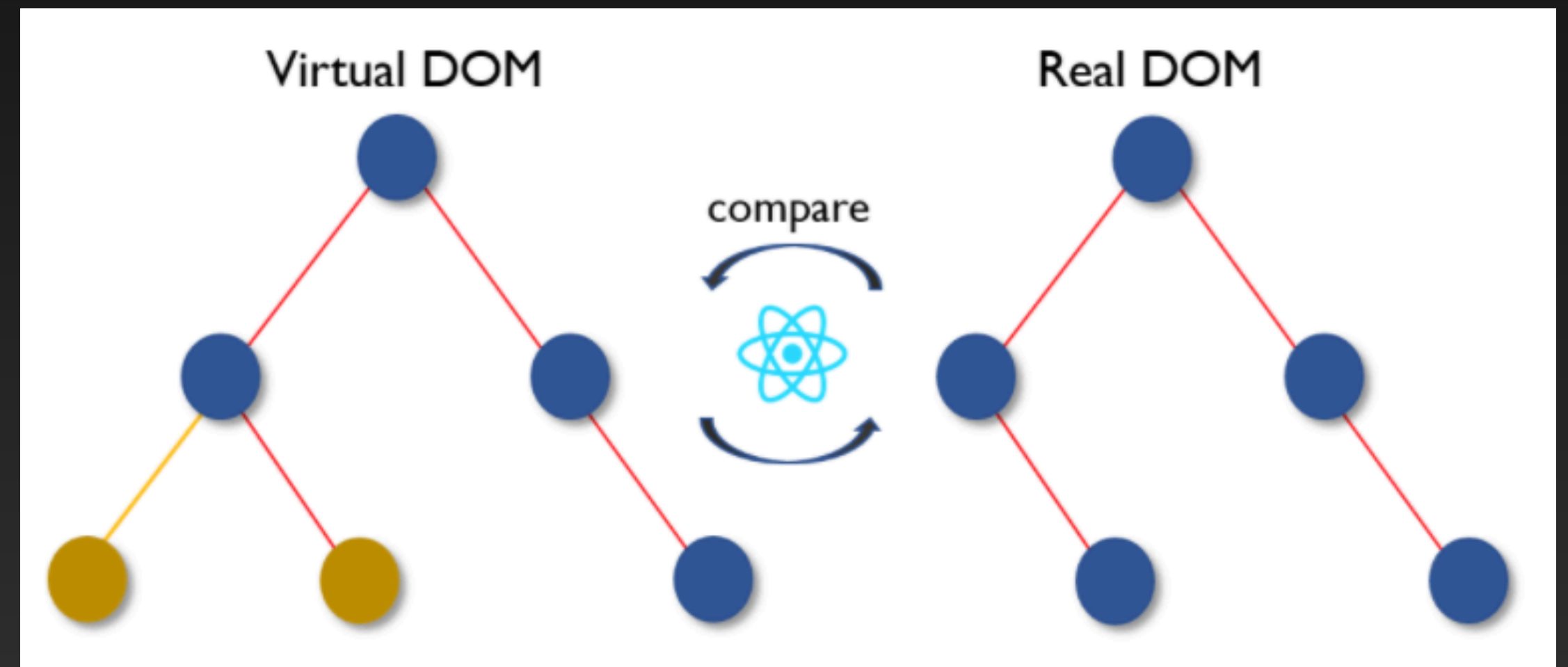
```
class HelloMessage extends React.Component {  
  render() {  
    return (  
      <div>  
        Hello {this.props.name}  
      </div>  
    );  
  }  
}  
  
ReactDOM.render(  
  <HelloMessage name="Taylor" />,  
  document.getElementById('hello-example')  
);
```

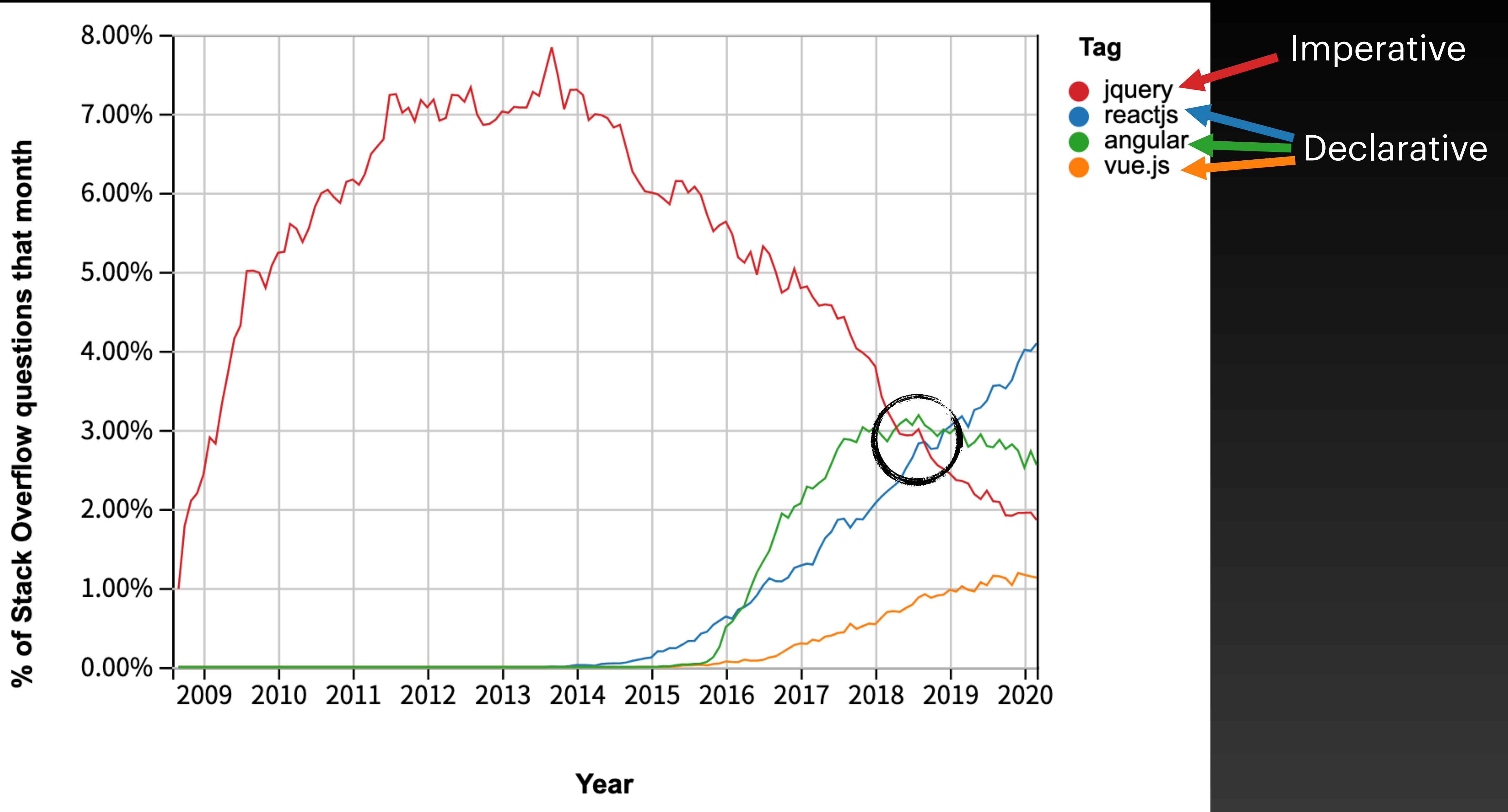


Rendering to Virtual DOM

```
class HelloMessage extends React.Component {  
  render() {  
    return (  
      <div>  
        Hello {this.props.name}  
      </div>  
    );  
  }  
}  
  
ReactDOM.render(  
  <HelloMessage name="Taylor" />,  
  document.getElementById('hello-example')  
);
```

render()





Paradigm Shifts

- 1990s - Static Webpages -> Dynamic Webpages
- 2000s - Dynamic HTML and Ajax (so called "Web 2.0")
- 2015~ - Imperative -> Declarative

PHP

```
<html>
<body>
<h1>User Greeting</h1>
<?php
// This will print whatever the user entered in
$name = filter_input(INPUT_GET, 'firstname',
FILTER_SANITIZE_STRING);
echo "Hello, ". $name . "!";
?>
</body>
</html>
```

Embed code in HTML in the backend

React

```
class HelloMessage extends React.Component {
  render() {
    return (
      <div>
        Hello {this.props.name}
      </div>
    );
  }
}

ReactDOM.render(
  <HelloMessage name="Taylor" />,
  document.getElementById('hello-example')
);
```

Embed HTML in code in the frontend

Paradigm Shifts

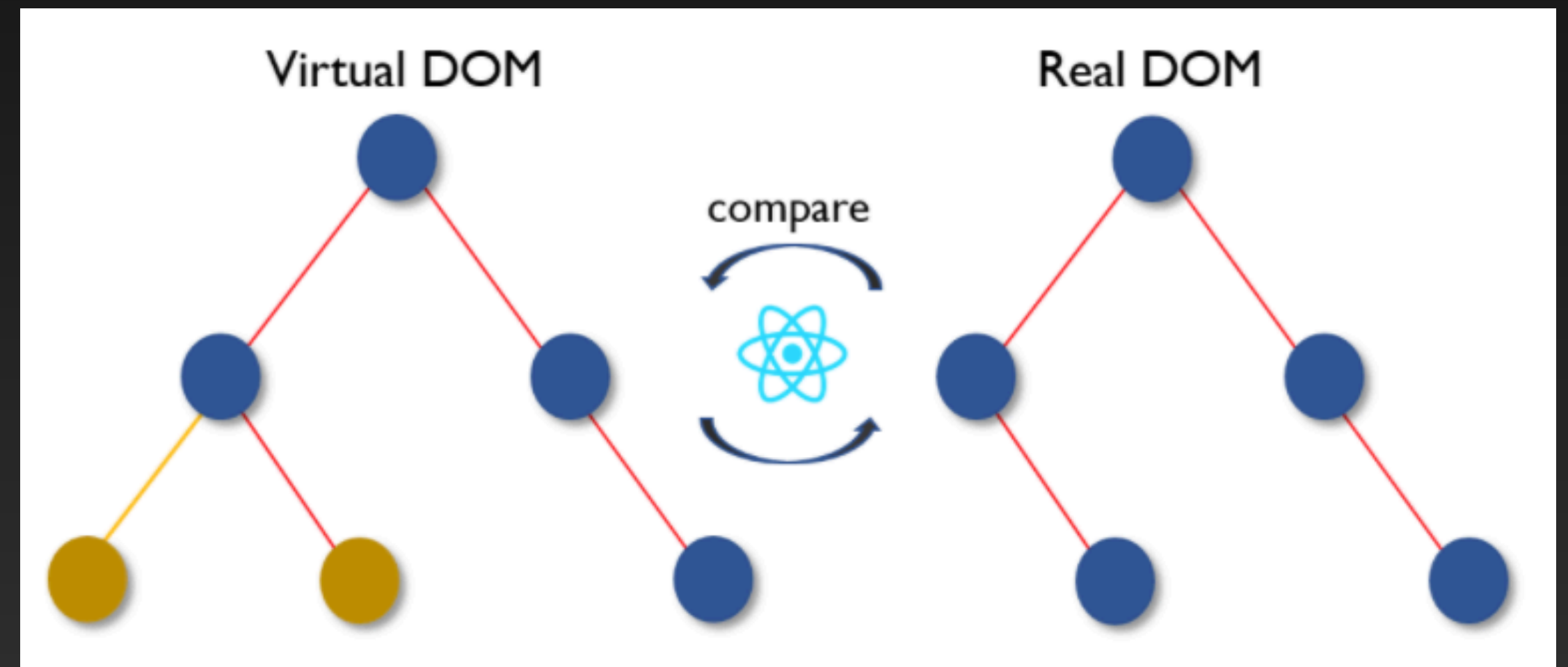
- Increase in computation power
 - Backend: C++ -> PHP -> Python/Ruby/NodeJS
 - Frontend: Imperative -> Declarative



Rendering to Virtual DOM

```
class HelloMessage extends React.Component {  
  render() {  
    return (  
      <div>  
        Hello {this.props.name}  
      </div>  
    );  
  }  
}  
  
ReactDOM.render(  
  <HelloMessage name="Taylor" />,  
  document.getElementById('hello-example')  
);
```

render()



Paradigm Shifts

- Increase in computation power
 - Backend: C++ -> PHP -> Python/Ruby/NodeJS
 - Frontend: Imperative -> Declarative
- Browsers became more standard compliant
 - ...and W3C are pushing out new standard specifications faster
 - Lowered effort for maintaining large frameworks

jQuery Browser Compatibility

```
<script language = "javascript" type = "text/javascript">
  <!--
  //Browser Support Code
  function ajaxFunction() {
    var ajaxRequest; // The variable that makes Ajax possible!

    try {
      // Opera 8.0+, Firefox, Safari
      ajaxRequest = new XMLHttpRequest();
    } catch (e) {

      // Internet Explorer Browsers
      try {
        ajaxRequest = new ActiveXObject("Msxml2.XMLHTTP");
      } catch (e) {

        try {
          ajaxRequest = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (e) {

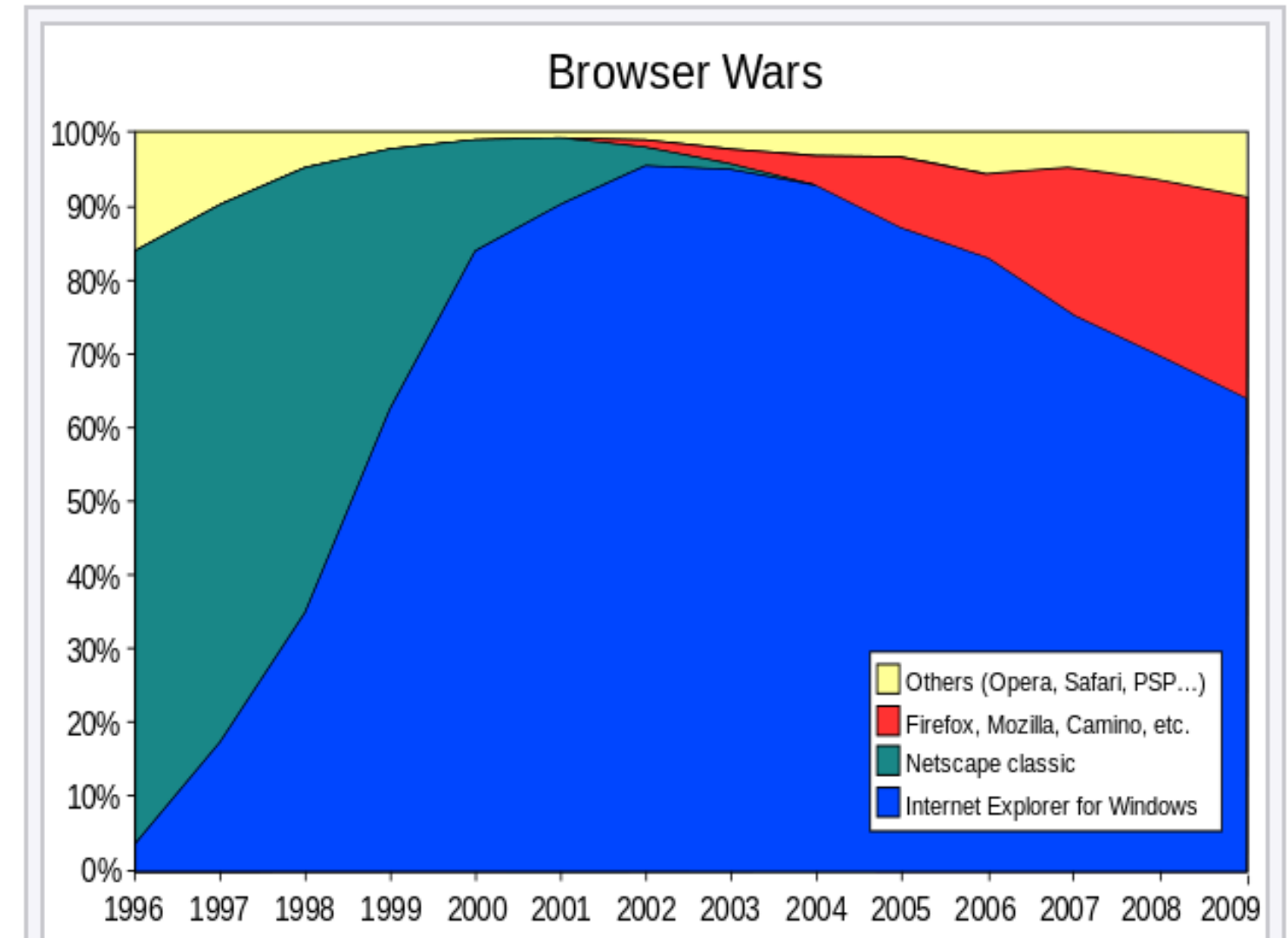
          // Something went wrong
          alert("Your browser broke!");
          return false;
        }
      }
    }
  }
}
```

jQuery.ajax(...)

First Browser War (1995–2001) [edit]

By mid-1995 the World Wide Web had received a great deal of attention in popular culture and the mass media. Netscape Navigator was the most widely used web browser and Microsoft had licensed Mosaic to create Internet Explorer 1.0, [7][8] which it had released as part of the Microsoft Windows 95 Plus! pack in August. [9]

Internet Explorer 2.0 was released as a free download three months later. Unlike Netscape Navigator, it was available to all Windows users free of charge, this also applying to commercial companies. [10] Other companies later followed suit and released their browsers free of charge. [11] Both Netscape Navigator and competitor products like InternetWorks, Quarterdeck Browser, InterAp, and WinTapestry were bundled with other applications to full Internet



Market share for several browsers between 1995 and 2010. Firefox was originally named "Phoenix", a name which implied that it would rise like a phoenix after Netscape Navigator was killed off by Microsoft.

Paradigm Shifts

- Increase in computation power
 - Backend: C++ -> PHP -> Python/Ruby/NodeJS
 - Frontend: Imperative -> Declarative
- Browsers became more standard compliant
 - ...and W3C are pushing out new standard specifications faster
 - Lowered effort for maintaining large frameworks
- New standards and APIs
- New software engineering tools
 - Transpiling Javascript, Static type-checking, package management, ...

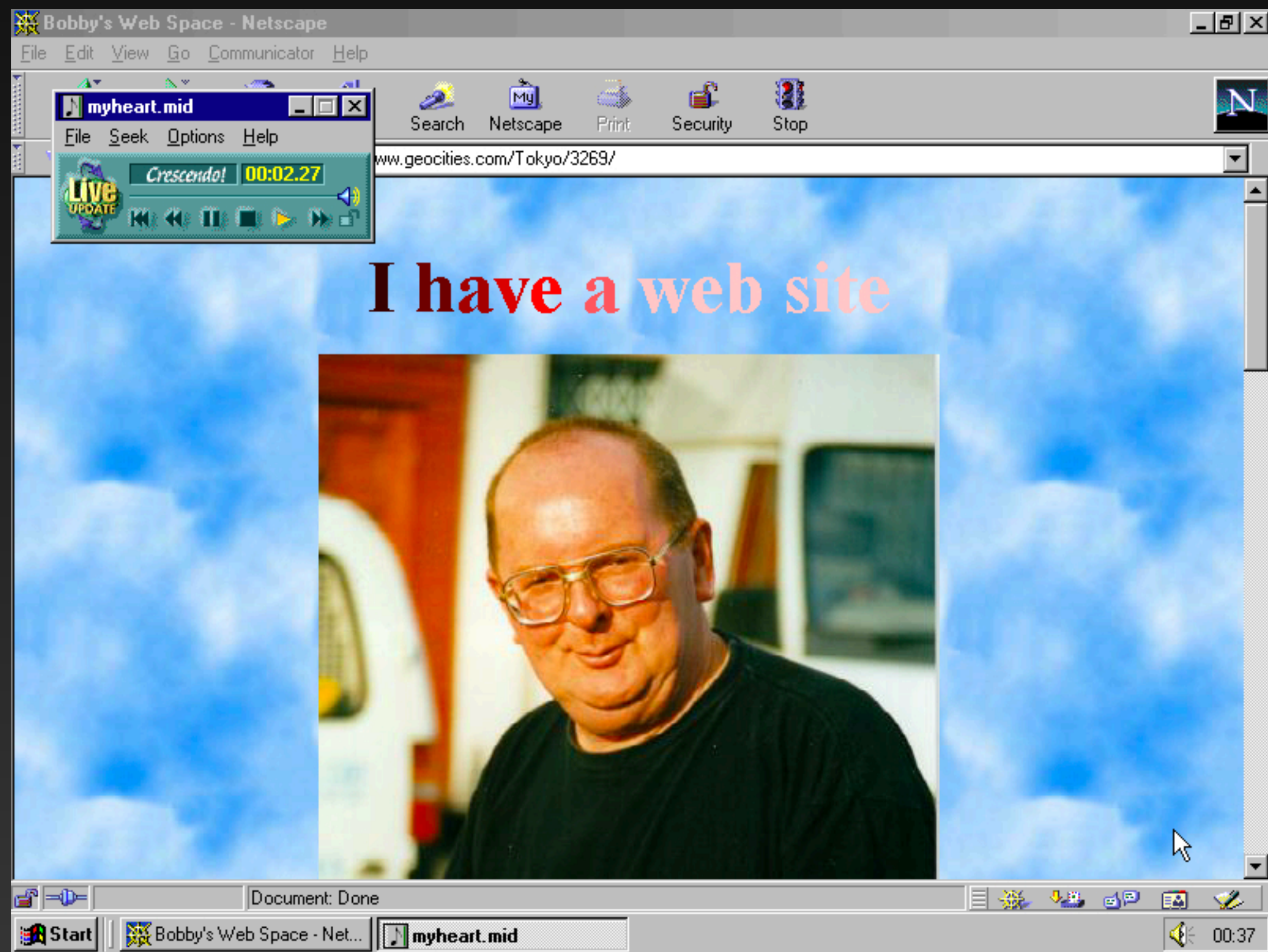
Going Forward

- React Native
 - Declarative framework for rendering native UI element on mobile
- Progressive Web App
 - Bring more native app characteristics to Web Apps

Modern Web Apps

Development & Toolkits

Nathan Hahn, April 16 2020



[Home](#) [Updates](#) [Stories](#) [Art](#) [Sonic](#) [Timelines](#) [Quotes](#) [About Me](#) [Links](#)

Chronos the Cat's Anime Adventure Episodes and Stories

[My Storylines](#) | [Other Episodes](#)

An "adventure" is a "user extensible" story. Like a Round Robin, different authors can take turns writing a story - however, unlike Round Robins, adventures branch, allowing different authors (or the same author if so inclined) to take the story in different directions. Anime Adventure is devoted to Anime fan fiction adventures (and crossovers including anime characters).

A list of most of the episodes I've written can be found [here](#).

[ADD PEOPLE TO THIS PROJECT](#)

[Go to project settings](#)

Resources

- App Engine
1 version
- Compute Engine
5 instances
- Storage
4 buckets
- Cloud Functions
45 functions
- BigQuery
2 datasets
- SQL
1 instance

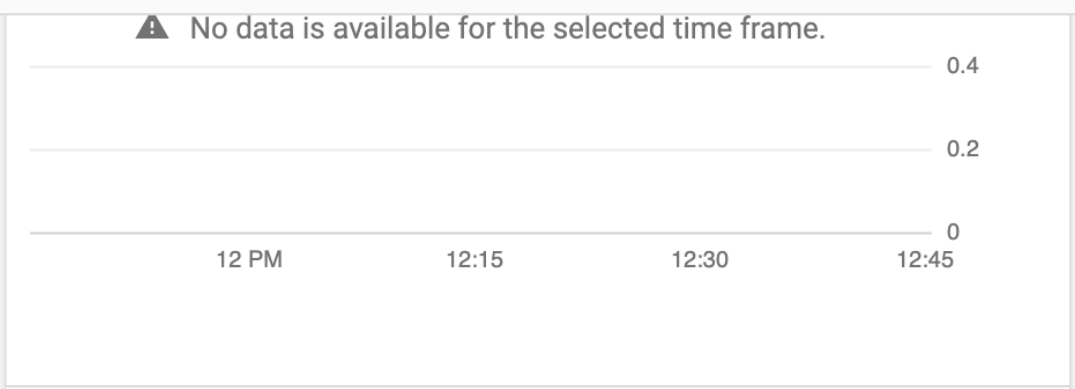
Trace

No trace data from the past 7 days

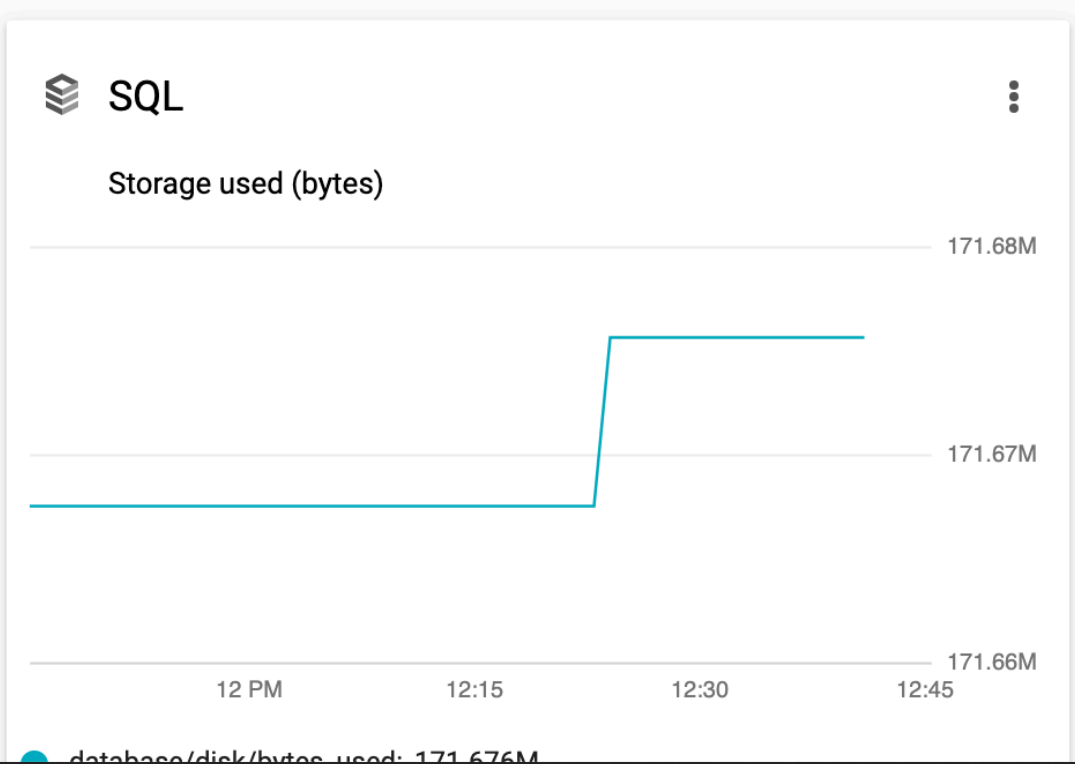
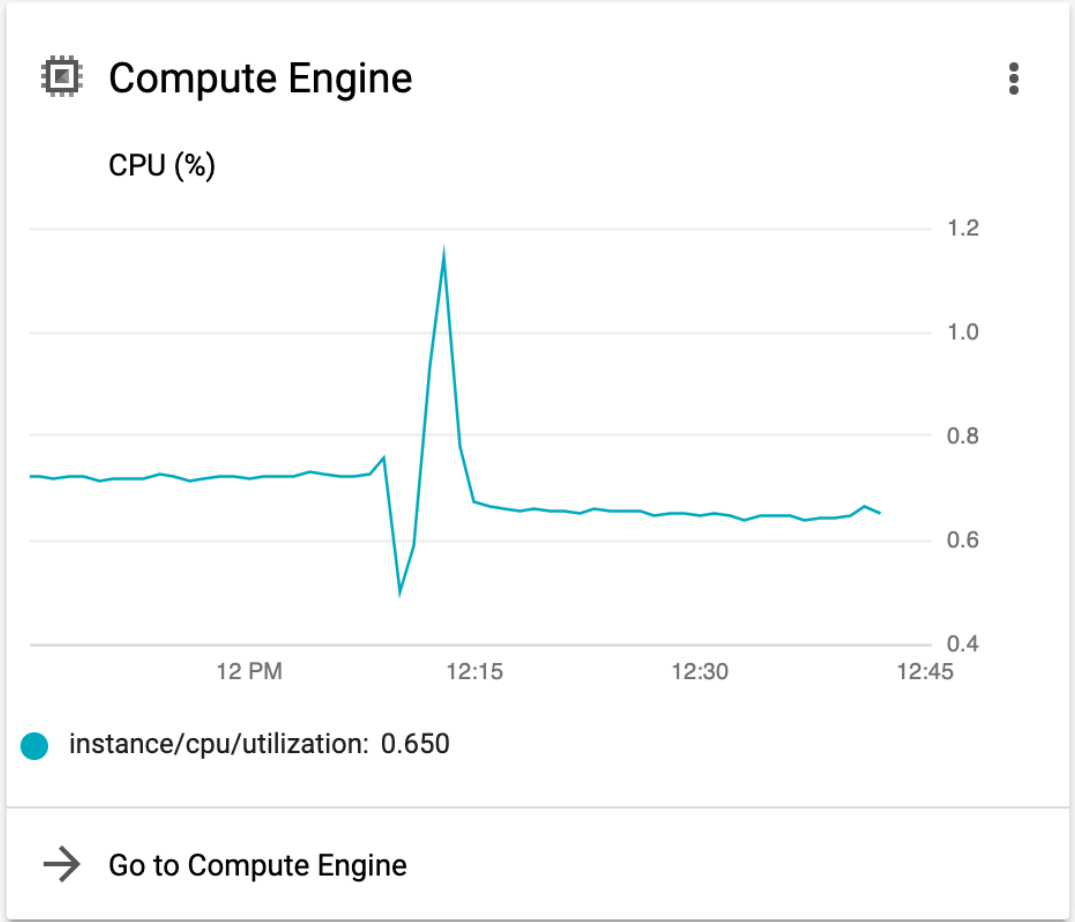
[Get started with Stackdriver Trace](#)

Getting Started

- API Explore and enable APIs
- Deploy a prebuilt solution
- Add dynamic logging to a running application
- Monitor errors with Error Reporting
- Install the Cloud SDK



[Go to the App Engine dashboard](#)



Billing

Estimated charges USD \$0.00
For the billing period Apr 1 - 16, 2020

[View detailed charges](#)

Error Reporting

Top errors in last 24 hours

- 1 Error
onTimeout()(timers.js)

[Go to Error Reporting](#)

News

- Kotlin, meet gRPC: a new open-source project for modern apps
45 minutes ago
- Protect your running VMs with new OS patch management service
45 minutes ago
- How Anthos GKE surge upgrades improve operational efficiency
1 day ago

[Read all news](#)

Documentation

- Learn about Compute Engine
- Learn about Cloud Storage
- Learn about App Engine



Build cross-platform desktop apps with JavaScript, HTML, and CSS

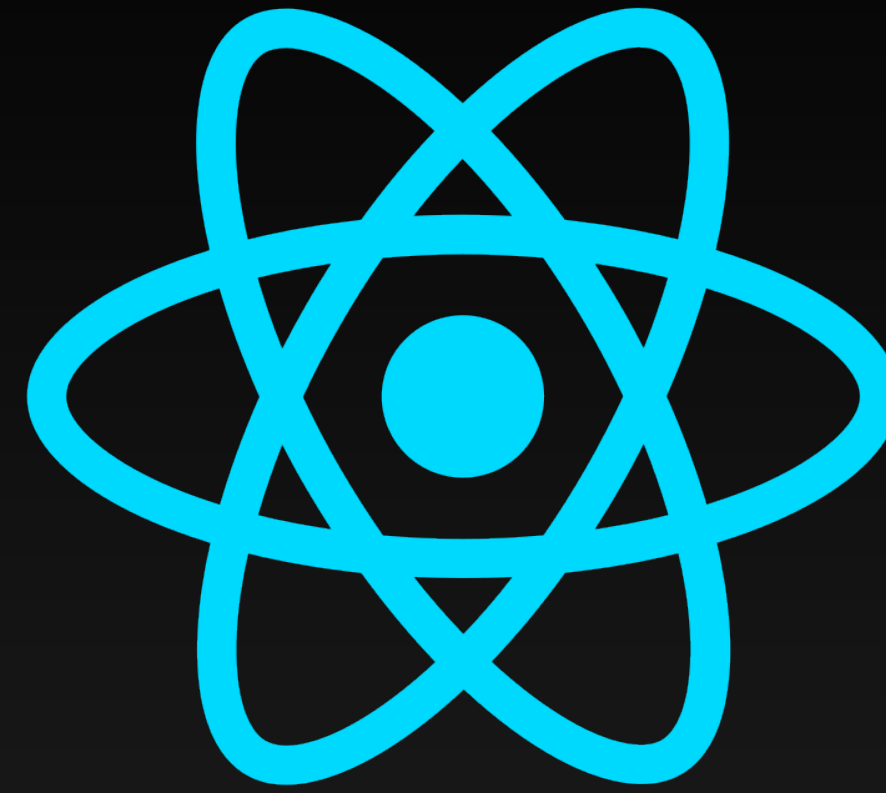
Releases

```
$ npm i -D electron@latest
# Electron 8.2.2
# Node 12.13.0
# Chromium 80.0.3987.163
```

```
$ npm i -D electron@beta
# Electron 9.0.0-beta.16
# Node 12.14.1
# Chromium 83.0.4102.3
```

```
$ npm i -D electron-nightly
# Electron 10.0.0-nightly.20200415
# Node 12.16.2
# Chromium 84.0.4115.0
```

It's easier than you think



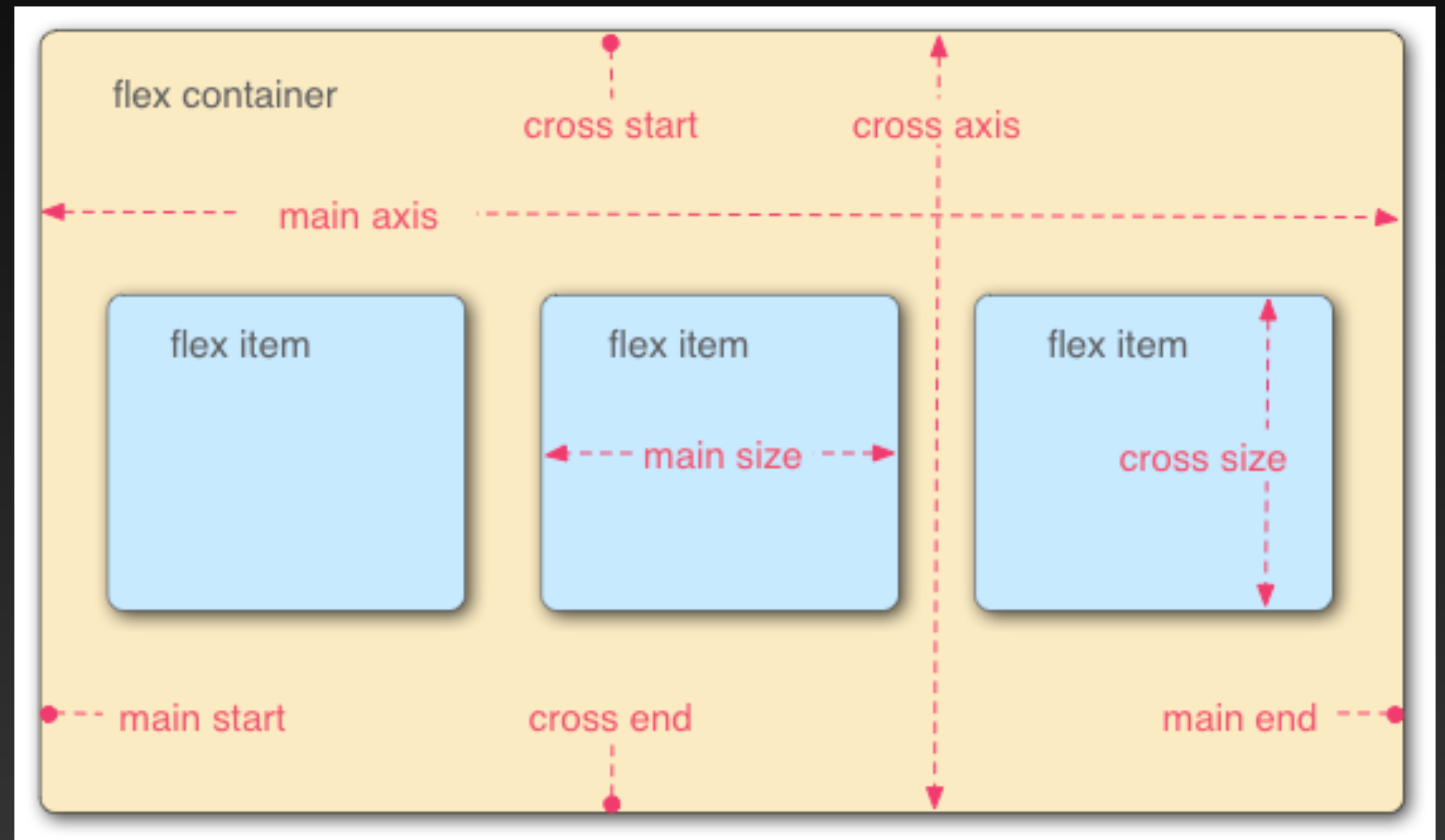


CSS3

- Media Queries
- **Transitions and Animations**
- **New Layout Types (Flexbox, CSS Grid)**
- Richer Unit Types (vw, vh, rem) and calc()
- **Graphics Transforms + Filters**
- Font Loading
- Box Shadows, Gradients, Rounded Corners
- Enhanced Selector Types

Flexbox

- Largely Constraint-Based
- Allows for significantly richer layout options



Transforms + Filters









rotate

scale

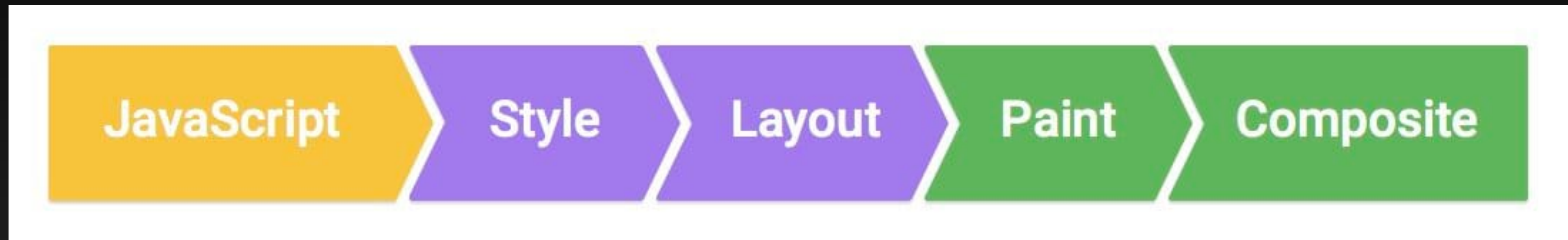
skew

translate

Filter Samples

Original	Grayscale	Sepia	Saturate
			
	<code>img {filter: grayscale(50 %);}</code>	<code>img {filter: sepia(50 %);}</code>	<code>img {filter: saturate(60 %);}</code>
Invert	Opacity	Brightness	Contrast
			
<code>img {filter: invert(90 %);}</code>	<code>img {filter: opacity(50 %);}</code>	<code>img {filter: brightness(50 %);}</code>	<code>img {filter: contrast(50 %);}</code>

Aside: Browser Rendering



Note: Layout and Paint can be skipped based on the CSS properties changed: i.e. shadows, color, or transforms

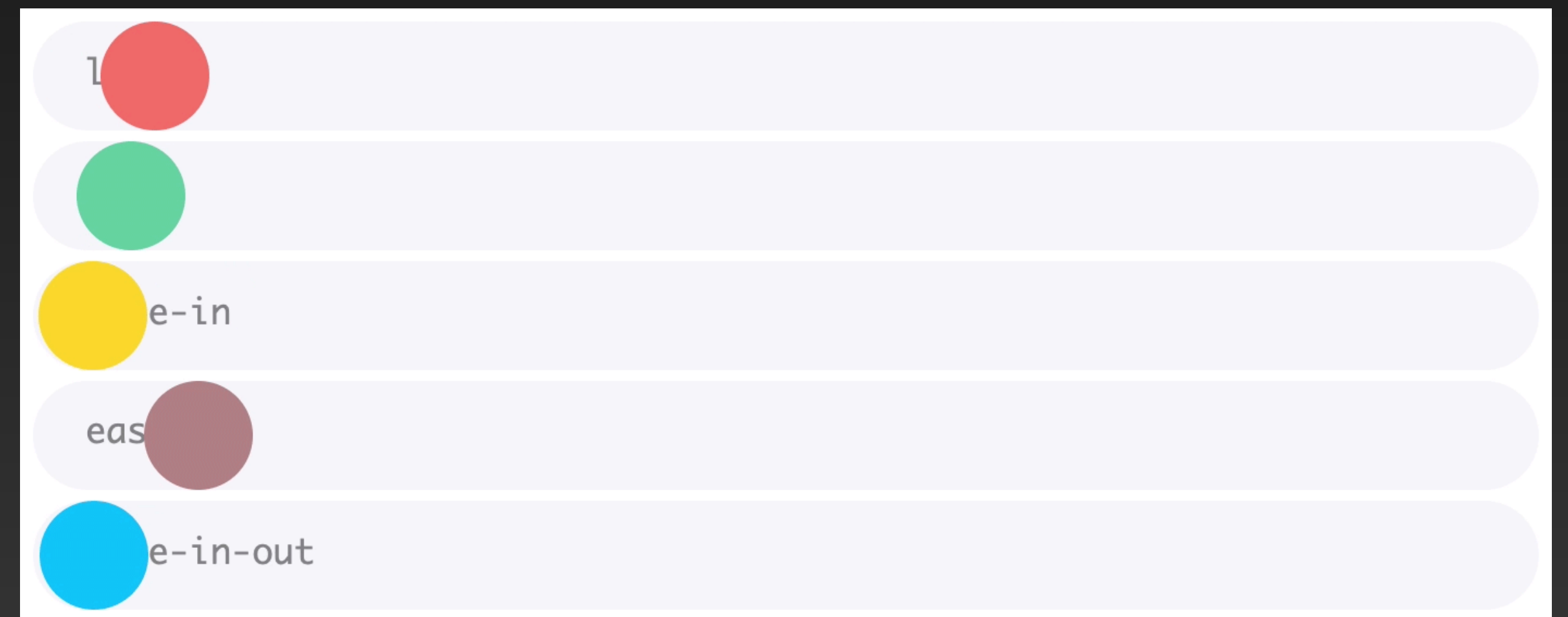
Animations

Provide a consistent, simplified interface for animating certain CSS properties

Transitions

```
transition: [*property] [*transition-duration]  
[transition-timing-function] [transition-delay];
```

```
transition: background-color 2s ease-out 0.5s;
```



Animations

Keyframes



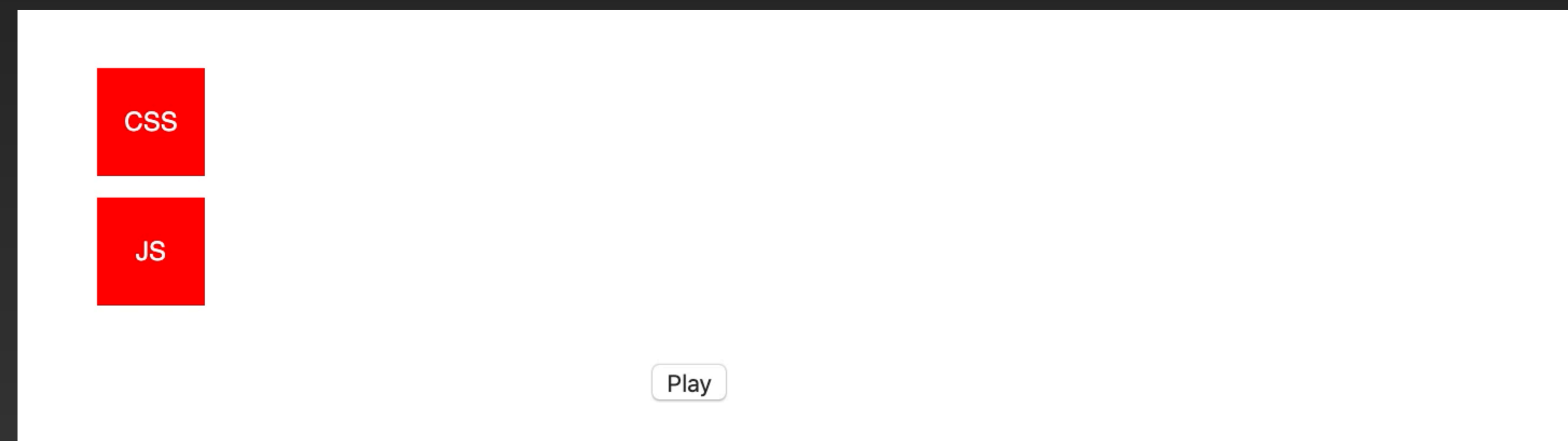
```
@keyframes bounce {  
  0% {  
    transform: scale(0.1);  
    opacity: 0;  
  }  
  60% {  
    transform: scale(1.2);  
    opacity: 1;  
  }  
  100% {  
    transform: scale(1);  
  }  
}  
.ball {  
  animation: bounce 2s infinite;  
}
```


Animations

They in most cases aren't actually more performant than JS scripted animations!

https://developer.mozilla.org/en-US/docs/Web/Performance/CSS_JavaScript_animation_performance

Except ... if they do not cause a layout change and have their own layer, they are completely GPU Animated



<https://www.smashingmagazine.com/2016/12/gpu-animation-doing-it-right/>

ES6

ES6 Javascript

- **Arrow Functions**
- Async (Promises, Await)
- **Modules**
- Variable Scoping (Let, Const)
- Template Literals
- Built-in Collection Functional Tools (map, reduce, etc.)
- Sets, Map, WeakMap
- Symbols
- **Classes**
- Proxies (Metaprogramming)

Classes

```
function Employee() {  
  this.name = '';  
  this.dept = 'general';  
}  
function Manager() {  
  Employee.call(this);  
  this.reports = [];  
}  
Manager.prototype =  
Object.create(Employee.prototype);  
Manager.prototype.constructor = Manager;
```

```
class Employee {  
  constructor() {  
    this.name = '';  
    this.dept = 'general';  
  }  
}  
class Manager extends Employee {  
  constructor() {  
    super();  
    this.reports = [];  
  }  
}
```

Modules

./modules/square.js

```
export const name = 'square';
```

```
export function draw(ctx, length, x, y, color) {  
  ctx.fillStyle = color;  
  ctx.fillRect(x, y, length, length);
```

```
  return {  
    length: length,  
    x: x,  
    y: y,  
    color: color  
  };  
}
```

... some other file

```
import { name, draw, reportArea, reportPerimeter } from './modules/square.js';
```

... or lazily

```
import('./modules/myModule.js')  
  .then((module) => {  
    // Do something with the module.  
  });
```


Arrow Functions

```
function square(x) { return x * x };
```

```
var square = x => x * x;  
square(7); // 49
```

```
range(1, 7).map(x => x * x); // -> [1, 4, 9, 16, 25, 36, 49]
```

```
const container = document.getElementById('container');
const btn = document.createElement('button');
btn.className = 'btn red';

btn.onclick = function(event) {
  if (this.classList.contains('red')) {
    this.classList.remove('red');
    this.classList.add('blue');
  } else {
    this.classList.remove('blue');
    this.classList.add('red');
  }
};

container.appendChild(btn);
```

HTML Templates

```
<template id="template">
  <script>
    const button =
      document.getElementById('click-me');
    button.addEventListener('click', event =>
alert(event));
  </script>
  <style>
    #click-me {
      all: unset;
      background: tomato;
      border: 0;
      border-radius: 4px;
      color: white;
      font-family: Helvetica;
      font-size: 1.5rem;
      padding: .5rem 1rem;
    }
  </style>
  <button id="click-me">Log click event</button>
</template>
```

```
const template =
document.getElementById('template');

document.body.appendChild(
  document.importNode(template.content, true)
);
```

What is React?

**A Declarative, Component-Based library
for building web applications**

Components

```
class Button extends React.Component{
  this.state = { color: 'red' }

  handleChange = () => {
    const color = this.state.color === 'red' ? 'blue' :
'red';
    this.setState({ color });
  }

  render() {
    return (<div>
      <button
        className={`btn ${this.state.color}`}
        onClick={this.handleChange}>
      </button>
    </div>);
  }
}
```

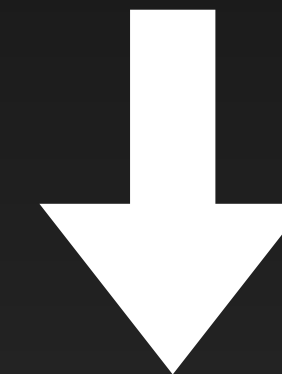
JSX

Technically syntactic sugar

Effectively allows us to write HTML templates and combine them with Javascript control flow

```
function getGreeting(user) {  
  if (user) {  
    return <h1>Hello, {formatName(user)}!</h1>;  
  }  
  return <h1>Hello, Stranger.</h1>;  
}
```

```
const element = (  
  <h1 className="greeting">  
    Hello, world!  
  </h1>  
);
```



```
const element = React.createElement(  
  'h1',  
  {className: 'greeting'},  
  'Hello, world!'  
);
```

Props + State

```
function Welcome(props) {  
  return <h1>Hello, {props.name}</h1>;  
}
```

```
function App() {  
  return (  
    <div>  
      <Welcome name="Sara" />  
      <Welcome name="Cahal" />  
      <Welcome name="Edite" />  
    </div>  
  );  
}
```

```
ReactDOM.render(  
  <App />,  
  document.getElementById('root')  
);
```

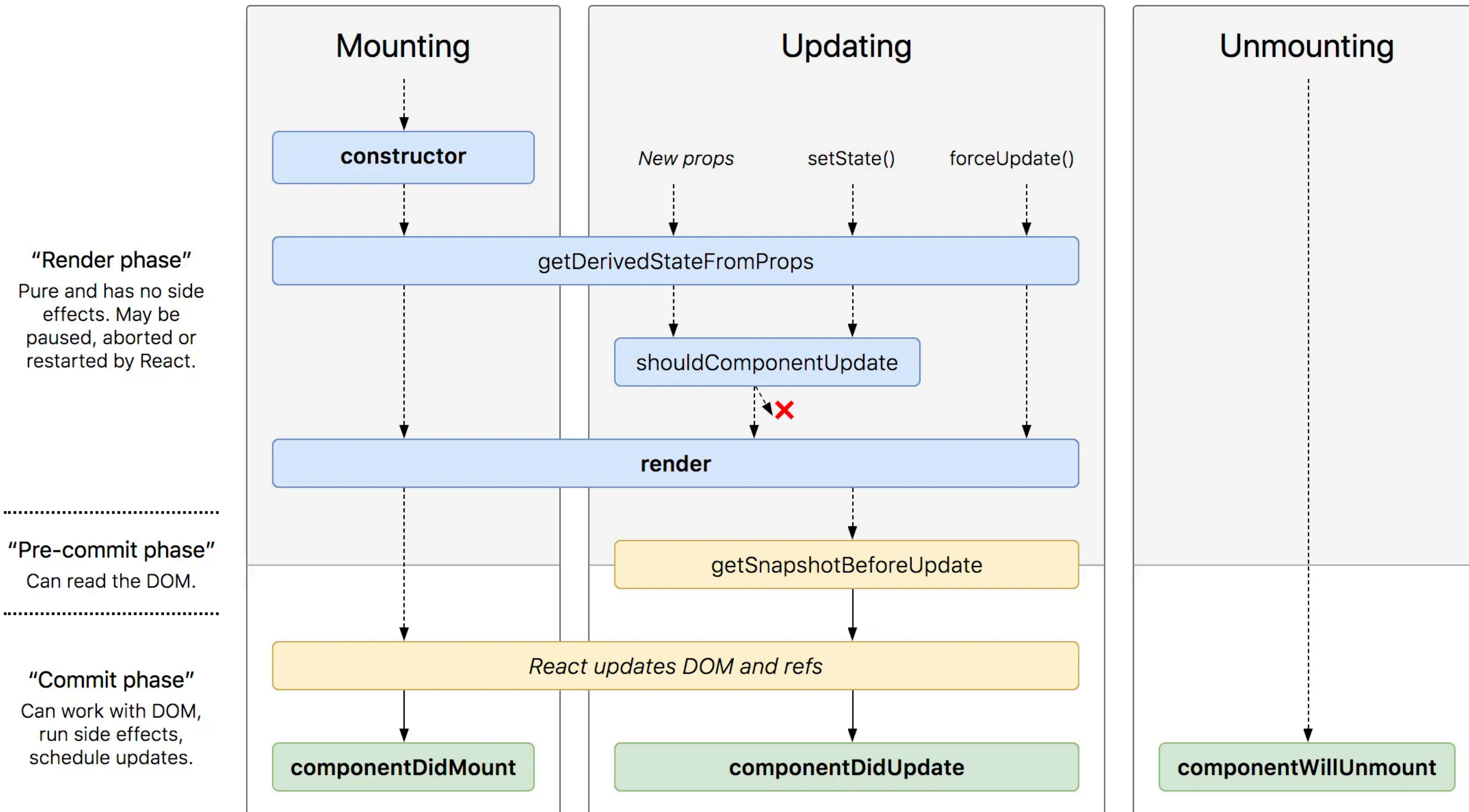
```
class Button extends React.Component {  
  this.state = { color: 'red' }
```

```
  handleChange = () => {  
    const color = this.state.color === 'red' ? 'blue' : 'red';  
    this.setState({ color });  
  }
```

```
  render() {  
    return (<div>  
      <button  
        className={`btn ${this.state.color}`}  
        onClick={this.handleChange}>  
      </button>  
    </div>);  
  }  
}
```

Lifecycle

React version 16.4 Language en-US



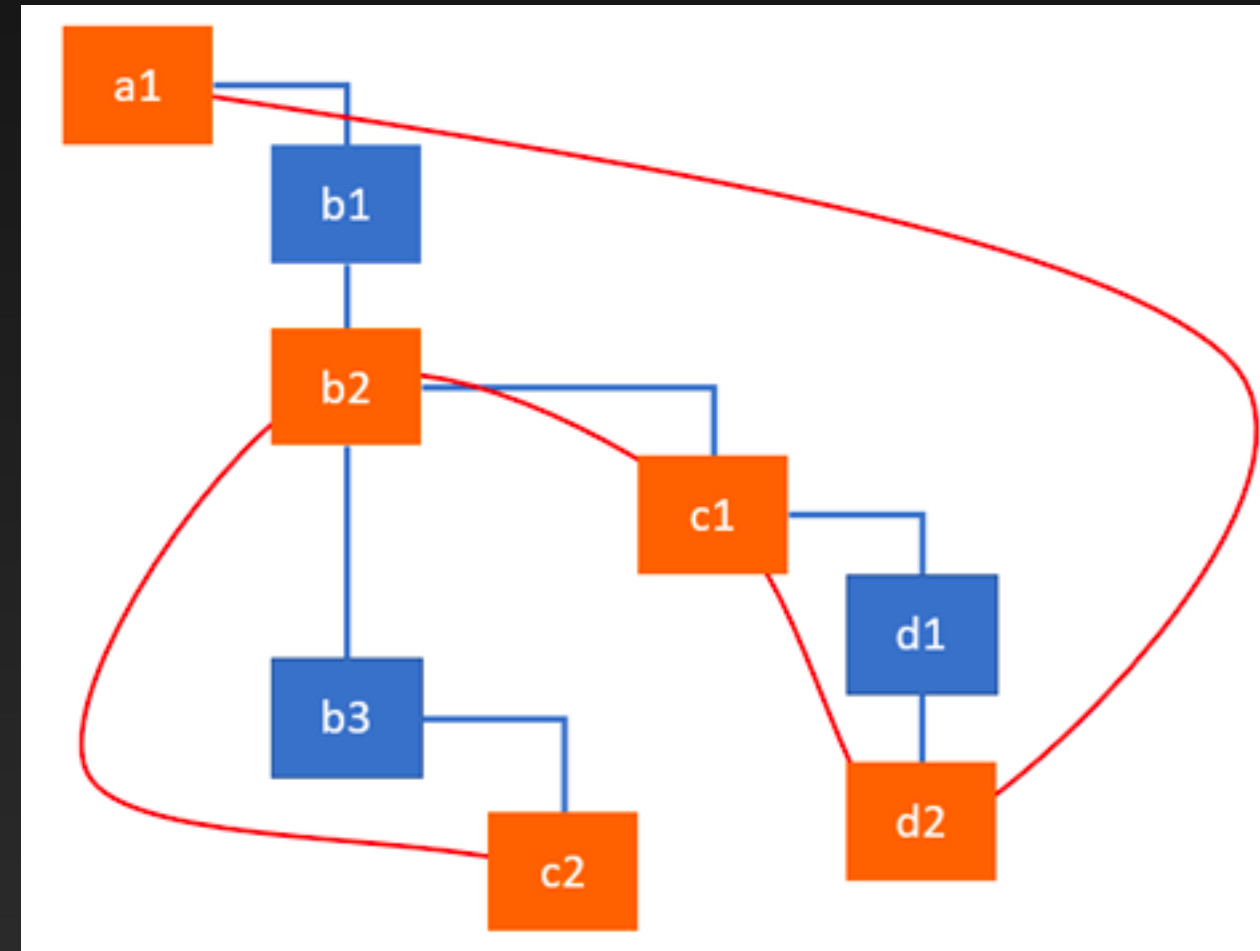
Reconciliation During Render

1. Skip to highest level setState
2. Look at the node — if it has changed type, rebuild the entire tree
3. Any changes to the DOM node are recorded
4. The the algorithm proceeds through the children
 1. If the number of children are changing programmatically, in order to appropriately preserve state for the child, a key should be used to identify it

Fiber

Render Phase

Produce a new Fiber tree based on the previous one.
Run render on node + children and create “working” tree



Commit Phase

Utilize the linked list to quickly update the DOM synchronously

<https://indepth.dev/inside-fiber-in-depth-overview-of-the-new-reconciliation-algorithm-in-react/>

Skipping the Render

Render can still be super expensive (depending on effects, logic, etc)

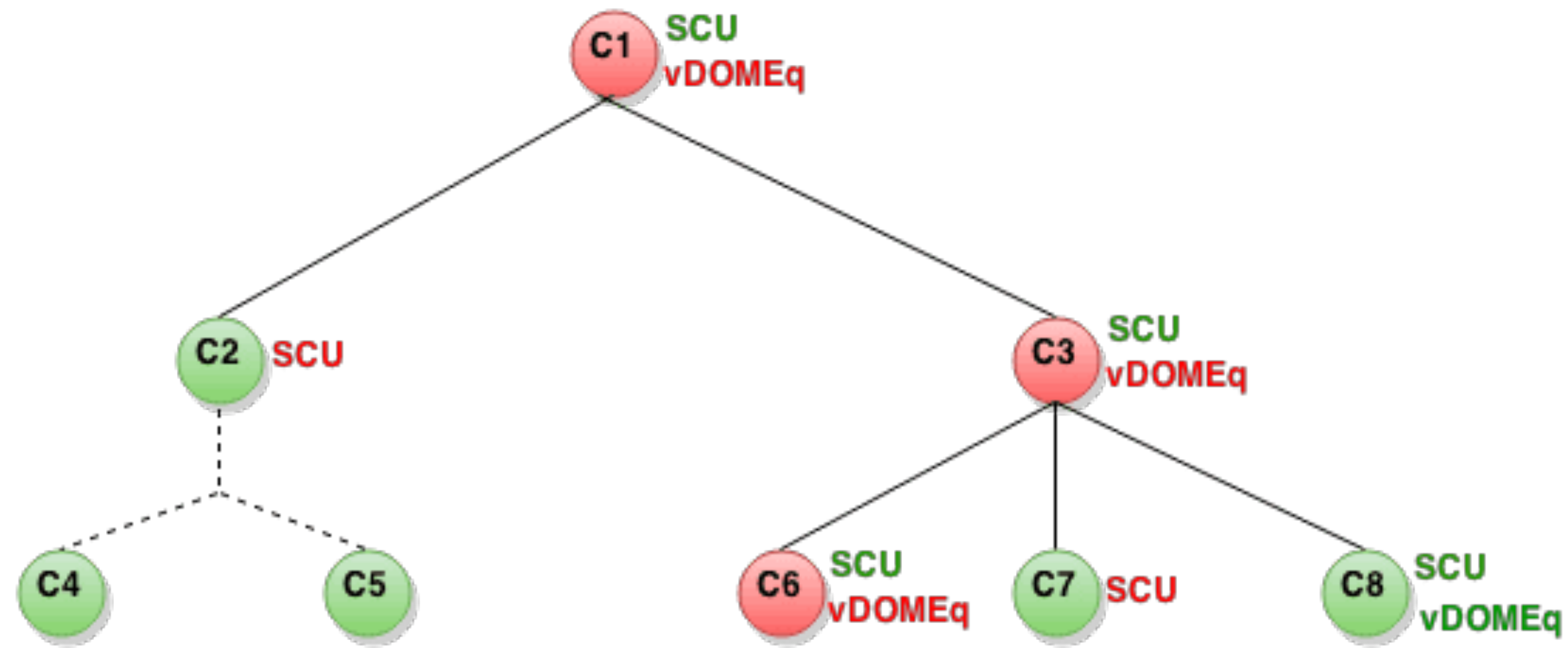
Ideally we want a short of a render as possible


```
shouldComponentUpdate(nextProps, nextState) {  
  return true;  
}
```


Ideally components should be “Pure” — only if their props or state has changed should they update

```
shouldComponentUpdate(nextProps, nextState) {  
  return this.props !== nextProps || this.state !== nextState  
}
```

Skipping the Render



 No Reconciliation needed

 Reconciliation needed

SCU shouldComponentUpdate?

SCU

vDOMEq are virtual DOMs equivalent?

vDOMEq

Moving Further into Declarative

Higher Order Components

A way to bind additional data and functionality to an existing component

```
// Take in a component as argument WrappedComponent
function withNameReact(WrappedComponent) {
  // And return a new anonymous component
  return class extends React.Component{
    render(){
      return <WrappedComponent
        name="React"
        {...this.props} />;
    }
  }
}
```

Moving Further into Declarative Functional Components

Simplify components even further — a component which response purely to its props (no state, no lifecycle)

```
function App() {  
  const greeting = 'Hello Function Component!';  
  return <Headline value={greeting} />;  
}  
function Headline(props) {  
  return <h1>{props.value}</h1>;  
}
```

Easy to shortcut rendering as well!

```
const MyComponent = React.memo(function MyComponent(props) {  
  /* render using props */  
});
```

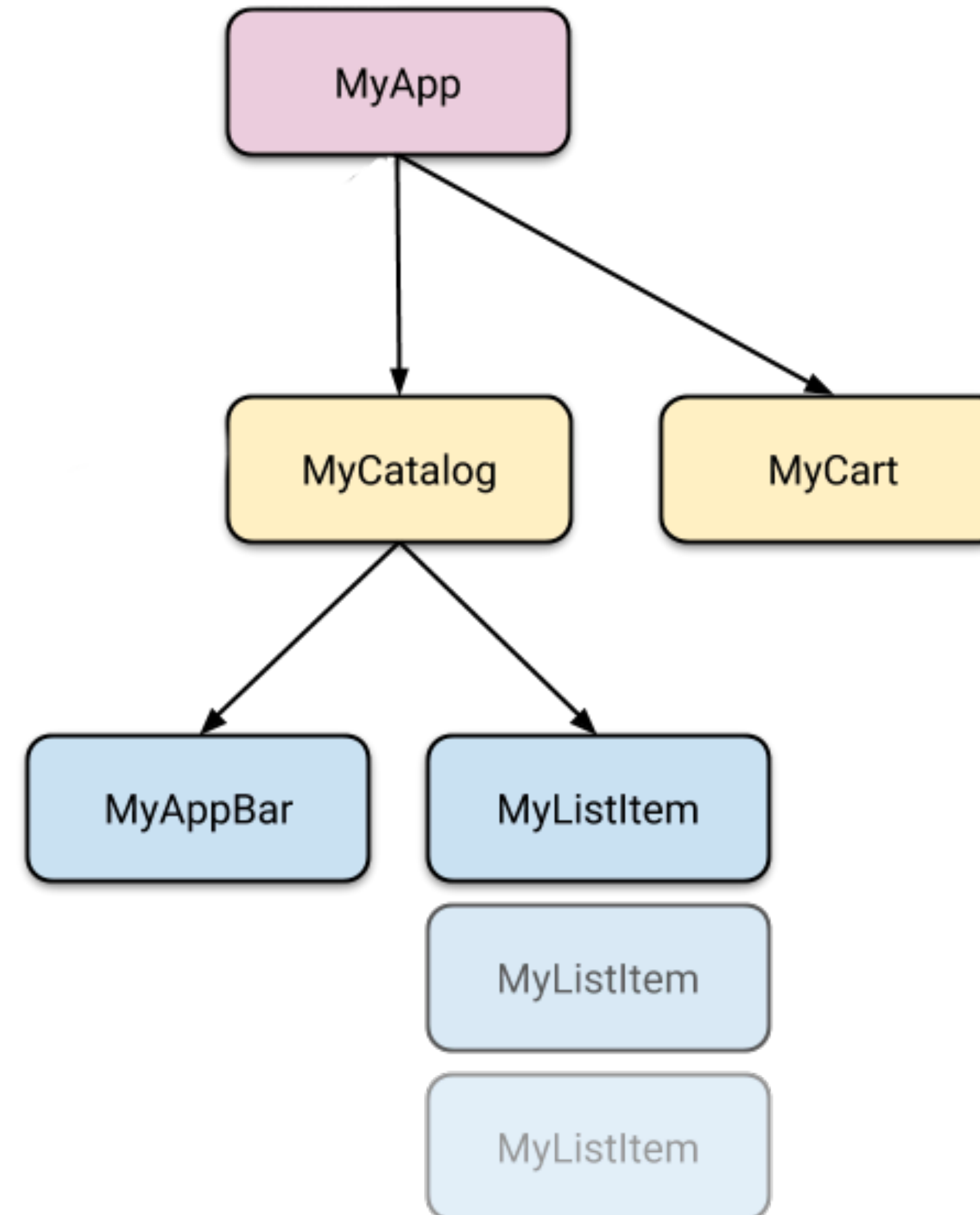
Moving Further into Declarative Hooks

Give functional components
state! But still have them be
as “Pure” as possible

```
function Example() {  
  // Declare a new state variable, which we'll call "count"  
  const [count, setCount] = useState(0);  
  
  useEffect(() => {  
    document.title = `You clicked ${count} times`;  
  }, [count]); // Only re-run the effect if count changes  
  
  return (  
    <div>  
      <p>You clicked {count} times</p>  
      <button onClick={() => setCount(count + 1)}>  
        Click me  
      </button>  
    </div>  
  );  
}
```

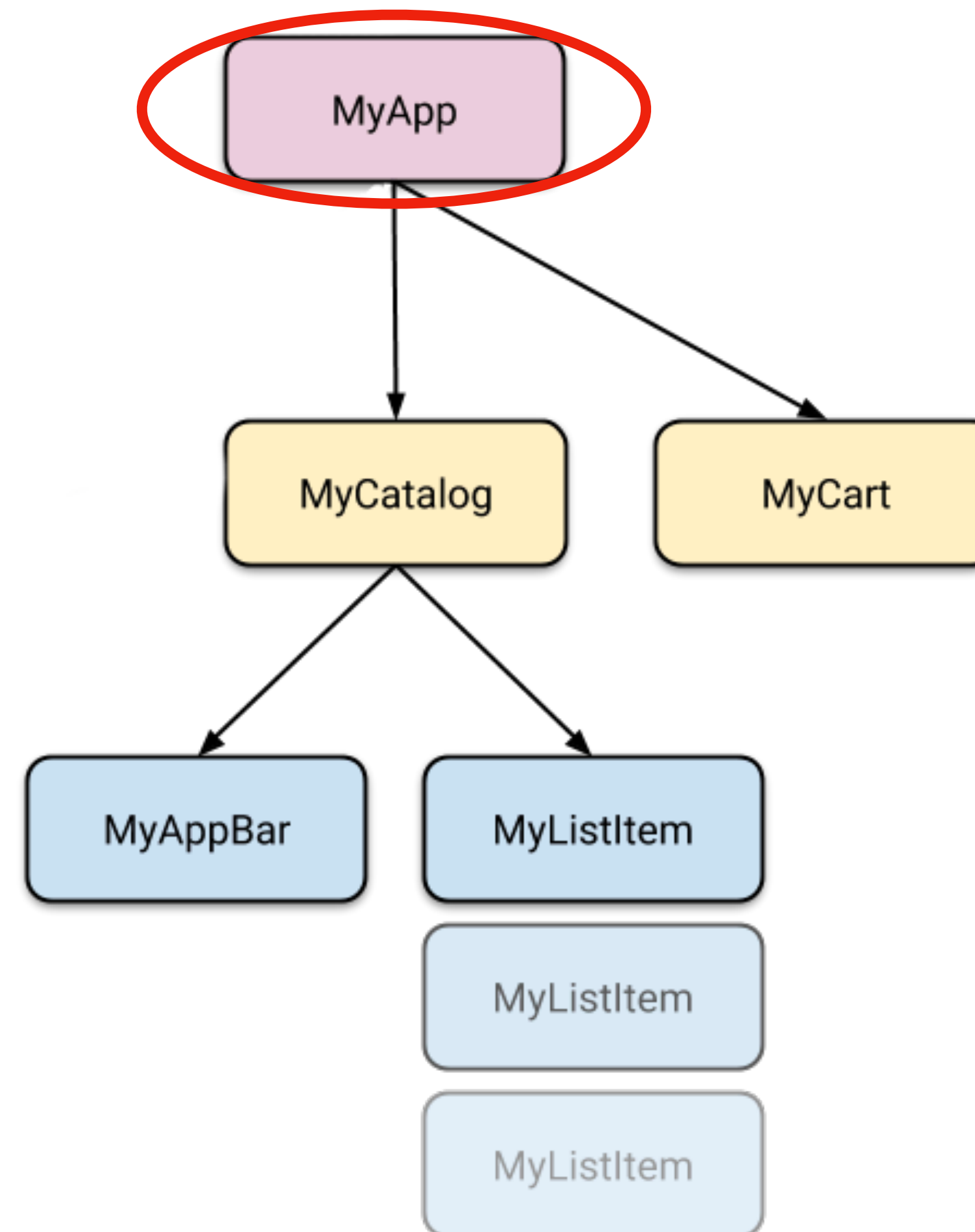
Lifting up State

Where should the state for what's in the cart go?



Lifting up State

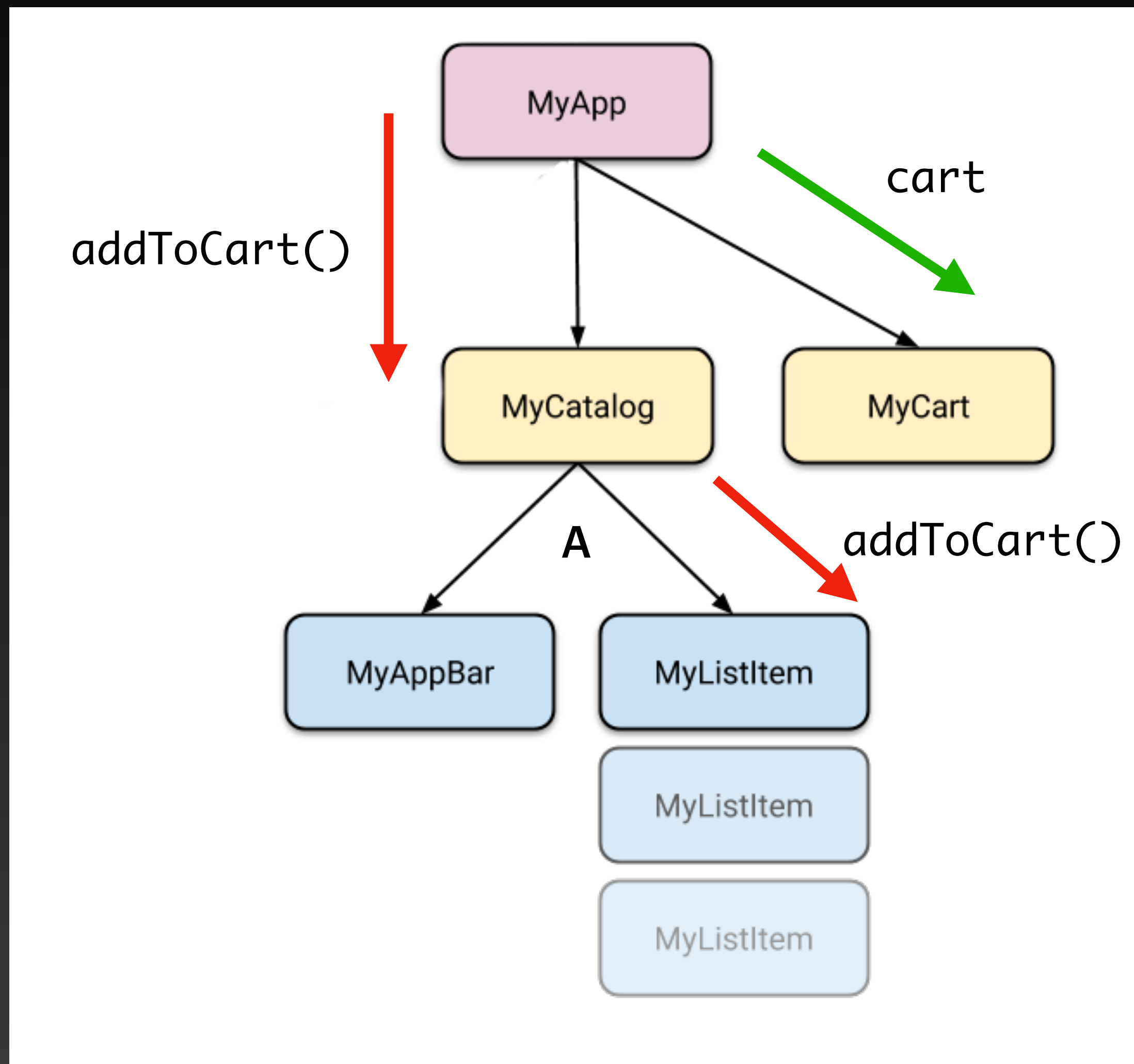
Where should the state for what's in the cart go?



Lifting up State

Well, then how do we add things to the cart and display them using “MyCart”?

```
function MyApp() {  
  const [cart, setCart] = useState([]);  
  
  const addToCart = (item) => {  
    setCart(c => [...c, item])  
  }  
}
```



Context

```
class App extends React.Component {  
  render() {  
    return <Toolbar theme="dark" />;  
  }  
}
```

```
function Toolbar(props) {  
  // The Toolbar component must take an extra "theme" prop  
  // and pass it to the ThemedButton. This can become painful  
  // if every single button in the app needs to know the theme  
  // because it would have to be passed through all components.  
  return (  
    <div>  
      <ThemedButton theme={props.theme} />  
    </div>  
  );  
}
```

```
class ThemedButton extends React.Component {  
  render() {  
    return <Button theme={this.props.theme} />;  
  }  
}
```

Context

```
// Context lets us pass a value deep into the component tree
// without explicitly threading it through every component.
// Create a context for the current theme (with "light" as the default).
const ThemeContext = React.createContext('light');
```

```
class App extends React.Component {
  render() {
    // Use a Provider to pass the current theme to the tree below.
    // Any component can read it, no matter how deep it is.
    // In this example, we're passing "dark" as the current value.
    return (
      <ThemeContext.Provider value="dark">
        <Toolbar />
      </ThemeContext.Provider>
    );
  }
}
```

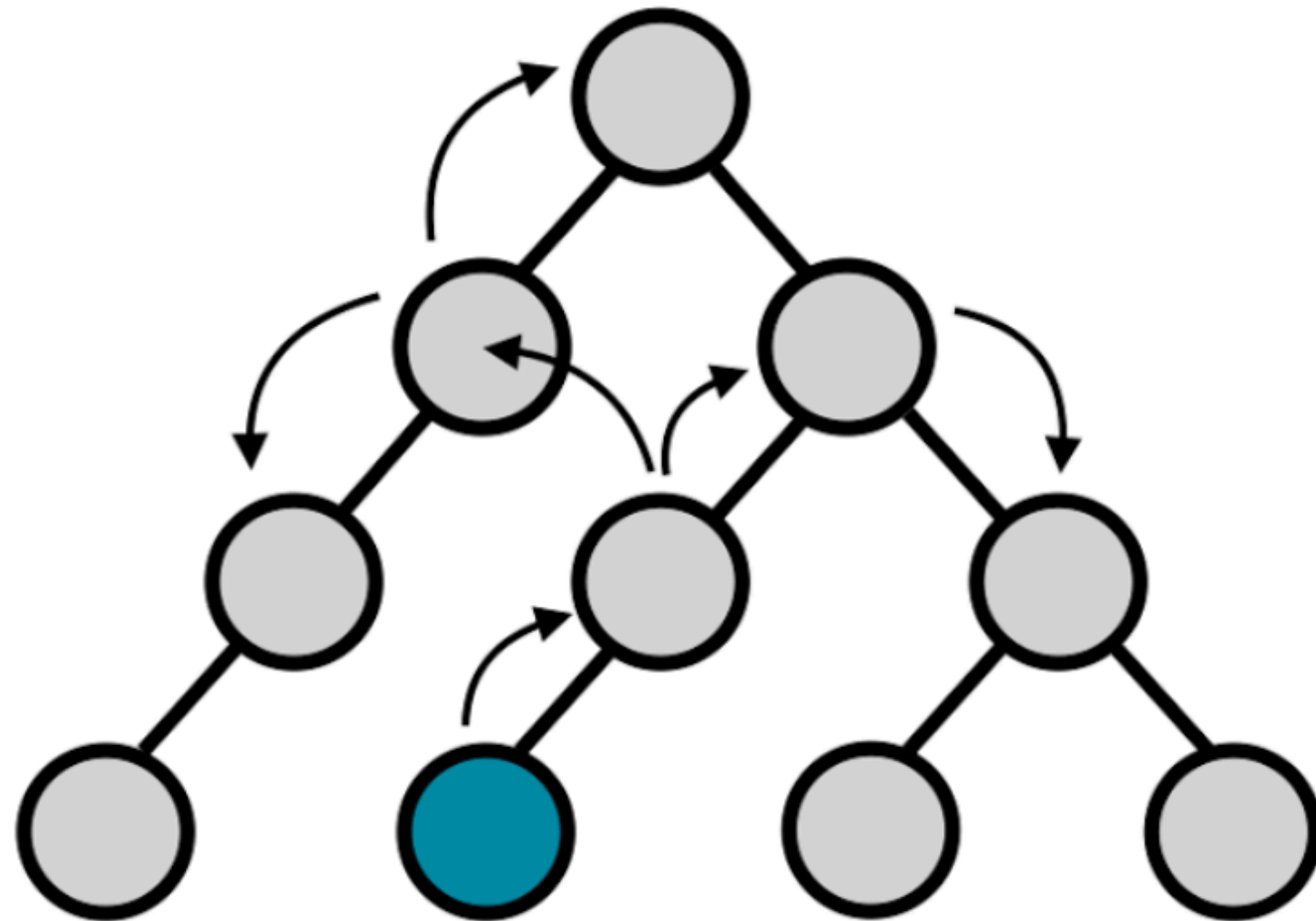
```
// A component in the middle doesn't have to
// pass the theme down explicitly anymore.
```

```
function Toolbar() {
  return (
    <div>
      <ThemedButton />
    </div>
  );
}
```

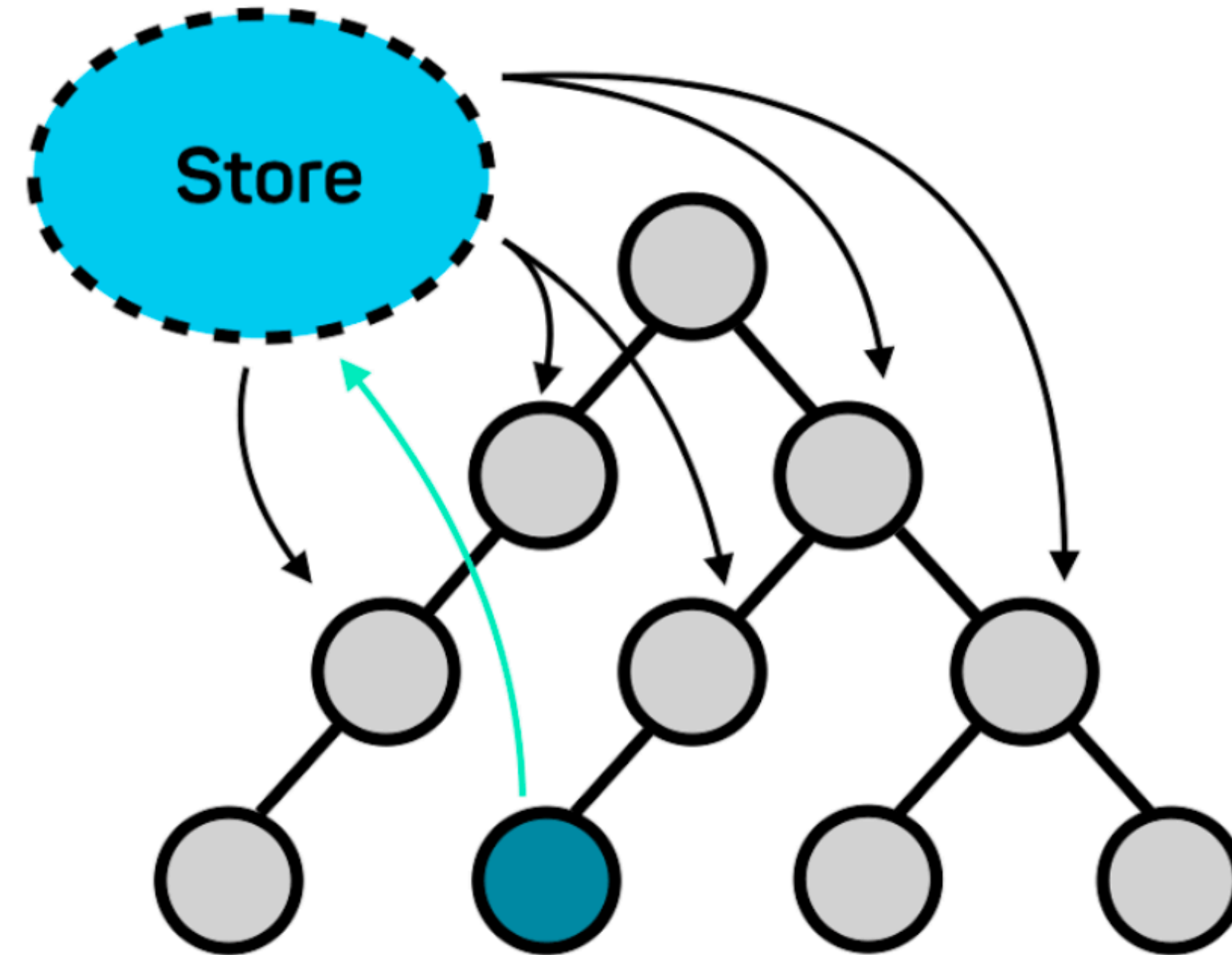
```
class ThemedButton extends React.Component {
  // Assign a contextType to read the current theme context.
  // React will find the closest theme Provider above and use its value.
  // In this example, the current theme is "dark".
  static contextType = ThemeContext;
  render() {
    return <Button theme={this.context} />;
  }
}
```

Redux

Without Redux

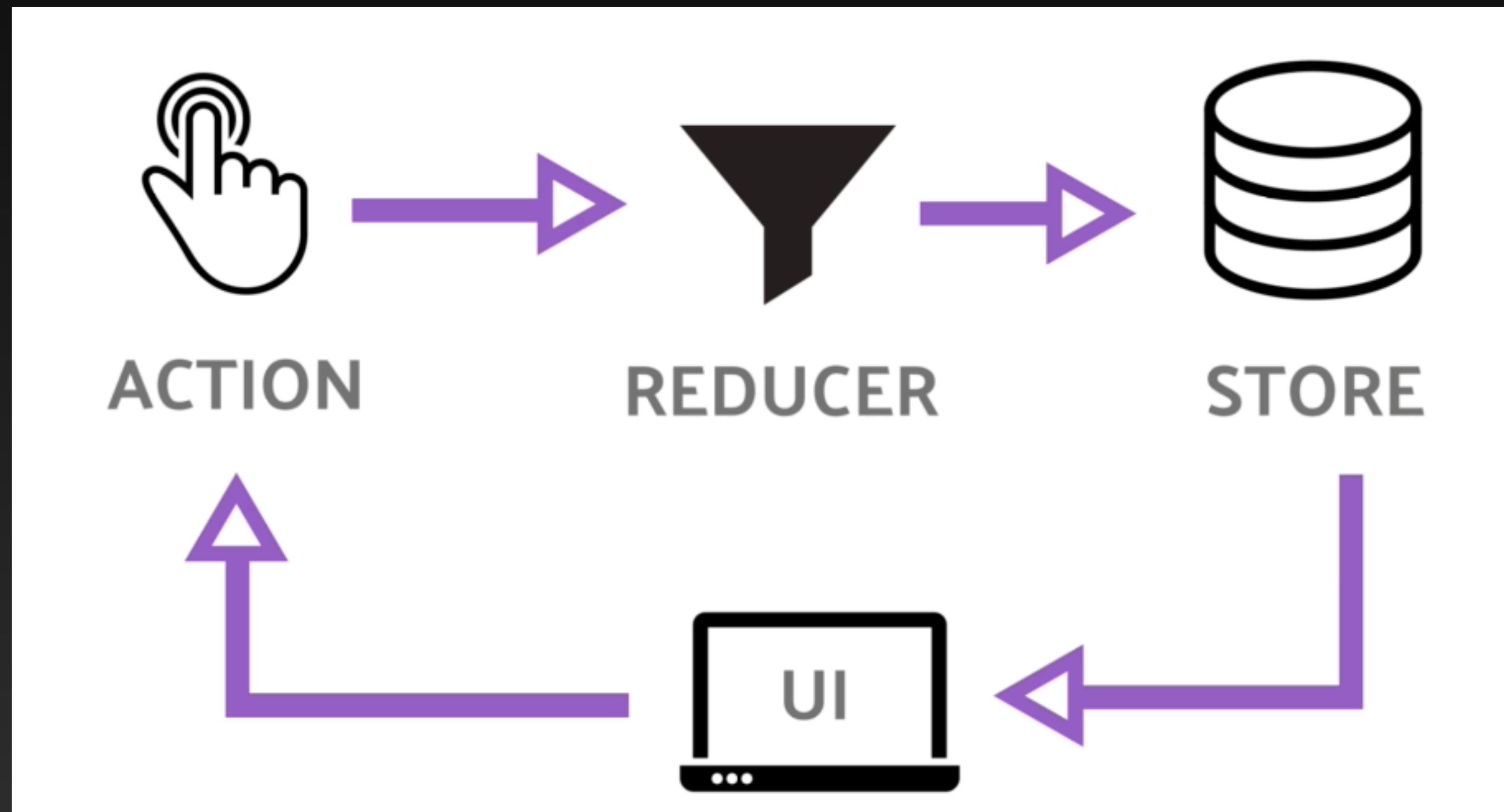


With Redux



Component initiating change

Redux



Redux

```
import React from 'react'  
import ReactDOM from 'react-dom'  
  
import { Provider } from 'react-redux'  
import store from './store'  
  
import App from './App'  
  
const rootElement = document.getElementById('root')  
ReactDOM.render(  
  <Provider store={store}>  
    <App />  
  </Provider>,  
  rootElement  
)
```

Redux

```
import { connect } from 'react-redux'  
import { increment, decrement, reset } from './actionCreators'
```

```
const Counter = function({counter, increment}) {  
  return <div onClick={() => increment()}>  
    {counter}  
  </div>  
}
```

```
const mapStateToProps = (state /*, ownProps*/) => {  
  return {  
    counter: state.counter  
  }  
}
```

```
const mapDispatchToProps = { increment, decrement, reset }
```

```
export default connect(  
  mapStateToProps,  
  mapDispatchToProps  
) (Counter)
```

Redux Hooks

```
import React, { useCallback } from 'react'
import { useDispatch, useSelector } from 'react-redux'

export const CounterComponent = () => {
  const counter = useSelector(state => state.counter)
  const dispatch = useDispatch()
  const incrementCounter = useCallback(
    () => dispatch({ type: 'increment-counter' }),
    [dispatch]
  )

  return (
    <div>
      <span>{counter}</span>
      <button onIncrement={incrementCounter}>Increment</button>
    </div>
  )
}
```

Immutability

Props should always be read-only! — except JS lets us change them anyways.



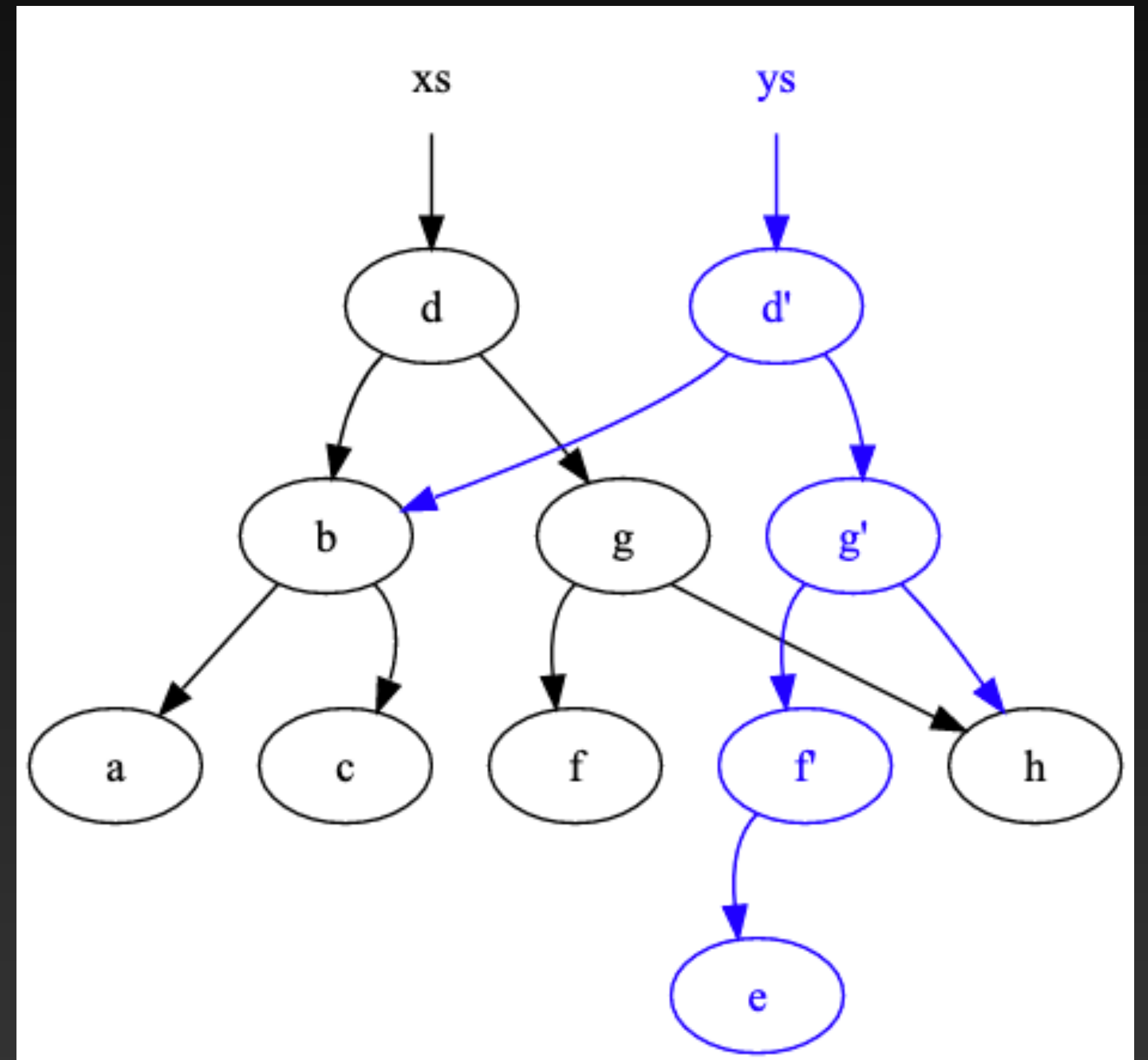
Immutability

```
let user = { name: "James Doe", location: "Lagos" }  
let newUser = user  
user.location = "Abia"  
console.log(newUser.location) // "Abia"
```

```
let state = { name: "James Doe", location: "Lagos" }  
state.location = "Abia"  
setState(state);  
//If Pure – Nothing will happen
```


Immutable.js

```
const { Map } = require("immutable");  
const map1 = Map({ a: 1, b: 2, c: 3 });  
const map2 = map1.set('b', 50);  
map1.get('b') + " vs. " + map2.get('b'); // 2 vs. 50  
map1 !== map2 //true
```



Immer

```
import produce from "immer"

const baseState = [
  {
    todo: "Learn typescript",
    done: true
  },
  {
    todo: "Try immer",
    done: false
  }
]

const nextState = produce(baseState, draftState => {
  draftState.push({todo: "Tweet about it"})
  draftState[1].done = true
})
```

```
// the new item is only added to the next state,
// base state is unmodified
expect(baseState.length).toBe(2)
expect(nextState.length).toBe(3)
```

```
// same for the changed 'done' prop
expect(baseState[1].done).toBe(false)
expect(nextState[1].done).toBe(true)
```

```
// unchanged data is structurally shared
expect(nextState[0]).toBe(baseState[0])
// changed data not (dûh)
expect(nextState[1]).not.toBe(baseState[1])
```

sort.svg

cards

amazon.svg

BrandGroup.tsx

CardGroup.tsx

DraggableCard.tsx

EntityCard.tsx

PlaceDetails.tsx

ProductDetails.tsx

list

AllOptions.tsx

Filter.tsx

Group.tsx

index.tsx

Menu.tsx

MyOptions.tsx

Sort.tsx

SourceDetails.tsx

SourceRow.tsx

Sources.tsx

SourceSummary.tsx

Tutorial.tsx

```
<GroupContainer
  style={{
    marginBottom: subEntities.length * 7 + 10
  }}
  className="cardGroup"
>
  <DraggableCard
    project={project}
    entity={entity.id}
    style={{ zIndex: 11 }}
  >
    <EntityCard
      entity={entity}
      project={project}
      projWindow={window}
      subEntities={subEntities}
    />
  </DraggableCard>
  {subEntities.map((child, idx) => (
    <ChildEntity
      key={child.id}
      style={{ top: idx * 6 - 4, left: idx * 3 + 3, zIndex: 10 - idx }}
    />
  ))}
</GroupContainer>
```

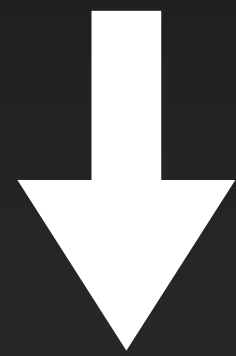


TypeScript

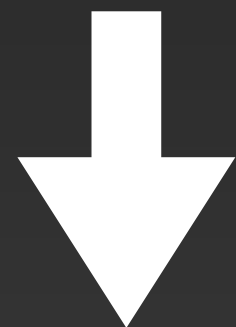
```
function greeter(person: string) {  
    return "Hello, " + person;  
}
```

```
let user = [0, 1, 2];
```

```
document.body.textContent = greeter(user);
```



tsc



greeter.js

error TS2345: Argument of type 'number[]' is not assignable to parameter of type 'string'.

TS function greeter(person: string) { Untitled-1 ●

```
1  function greeter(person: string) {  
2    return "Hello, " + person;  
3  }  
4  
5  let user = [0, 1, 2];  
6  
7  document.body.textContent = greeter(user);  
8
```

```
let user: number[]
```

Argument of type 'number[]' is not assignable to parameter of type 'string'. ts(2345)

[Peek Problem](#) No quick fixes available


```
enum AuthenticationMethod {  
  FACEBOOK = 'facebook',  
  EMAIL = 'email',  
  PHONE = 'phone'  
}  
  
interface UserProps extends DefaultProps {  
  name: string  
  nickname: string  
  authenticationMethod: AuthenticationMethod  
  likesCount: number  
}  
  
interface State {  
  isModalVisible: boolean  
}  
  
class User extends React.PureComponent<UserProps, State> {
```

Item.tsx

List.tsx

User Settings

```
1 import * as React from 'react';
2 import Item
3
4 export default
5   render() {
6     return (
7       <Item title={12} status='someString' />
8     );
9   }
10 }
11
```

[ts]

Types of property 'title' are incompatible.

Type 'number' is not assignable to type 'string'.

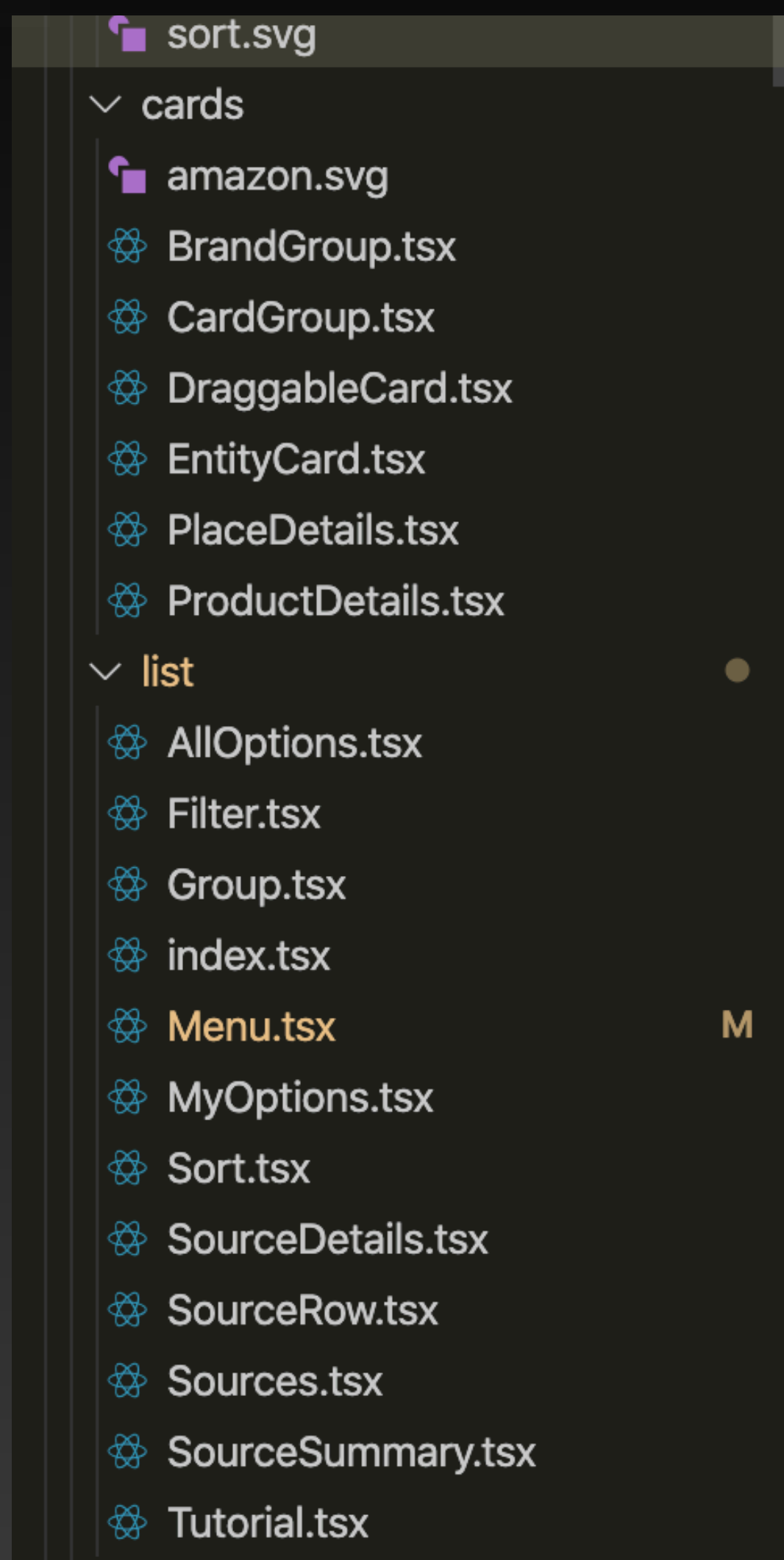
(JSX attribute) title: number

```
1  import * as React from 'react';
2  interface IProps {
3    title: string;
4    status: boolean;
5  }
6  export default class Item extends React.Component<IProps> {
7    render() {
8      return (
9        <div>
10       <h2>{this.props.}</h2>
11       </div>
12     );
13   }
14 }
```

- children?
- status (property) status: boolean ⓘ
- title



Webpack



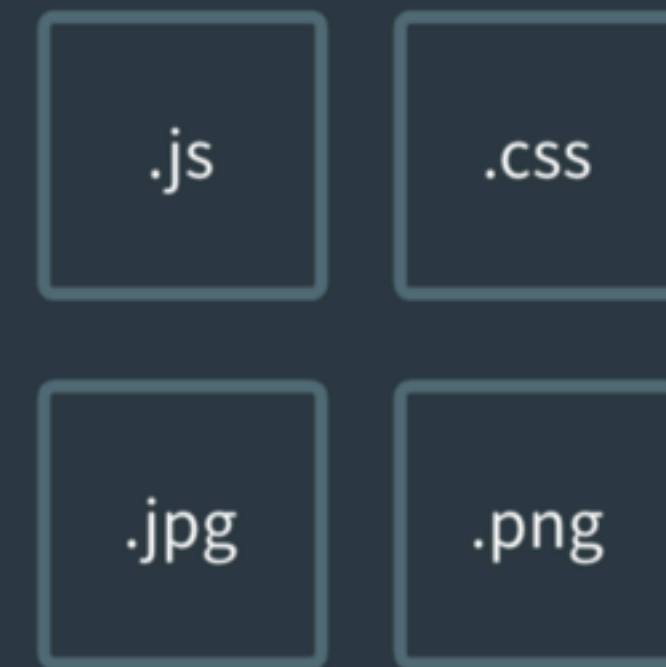
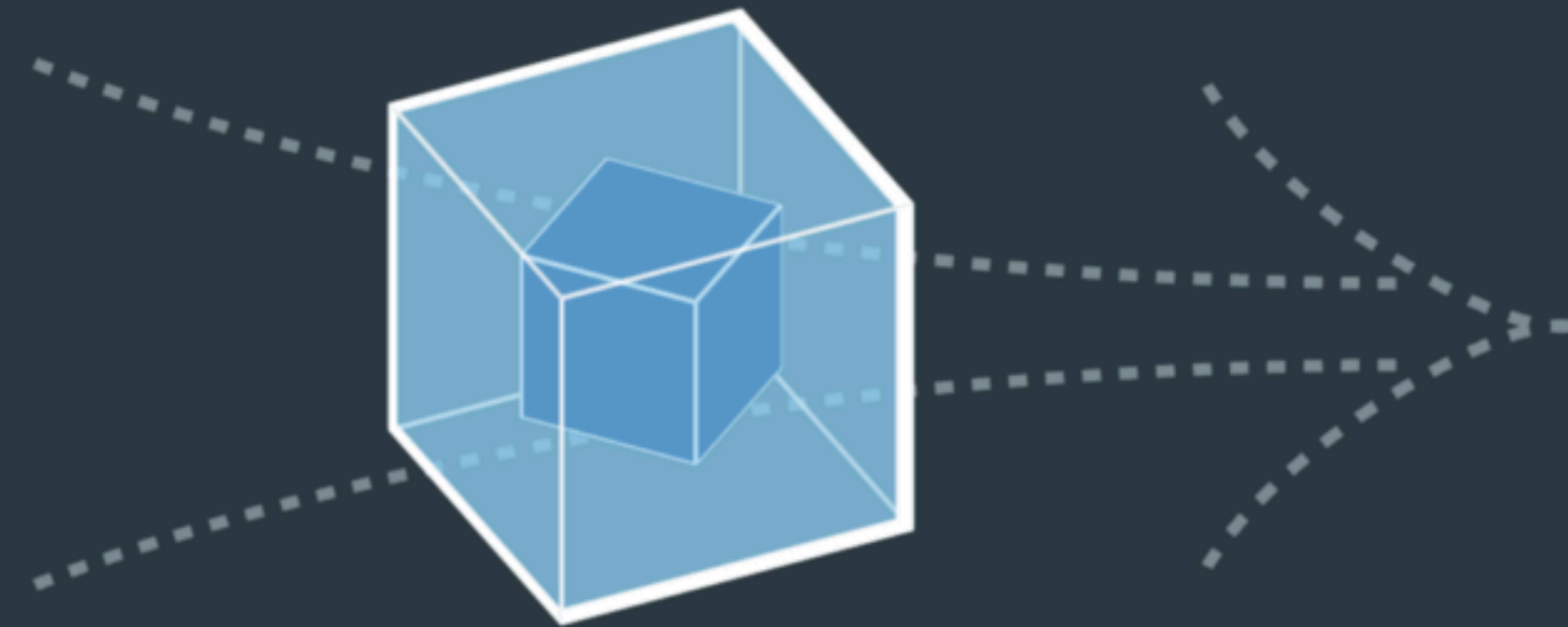
```
<script type="module" src="Sort.js"></script>  
<script type="module" src="Filter.js"></script>  
<script type="module" src="Group.js"></script>
```

```
•  
•  
•
```

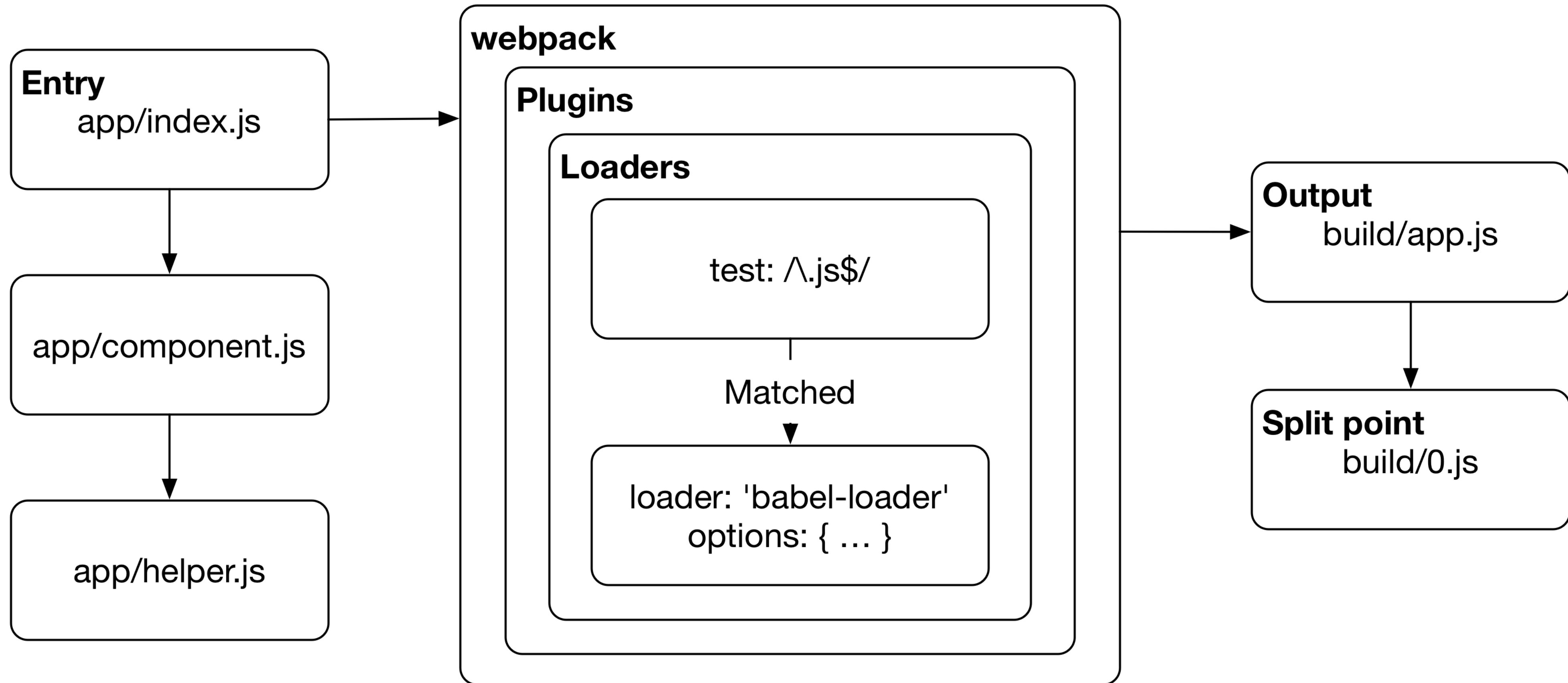
It's a Bundler



MODULES WITH DEPENDENCIES



STATIC ASSETS



Configuration

```
const webpack = require("webpack");

module.exports = {
  // Where to start bundling
  entry: {
    app: "./entry.js",
  },
  // Where to output
  output: {
    // Output to the same directory
    path: __dirname,

    // Capture name from the entry using a pattern
    filename: "[name].js",
  },
  // How to resolve encountered imports
  module: {
    rules: [
      {
        test: /\.css$/,
        use: ["style-loader", "css-loader"],
      },
      {
        test: /\.js$/,
        use: "babel-loader",
        exclude: /node_modules/,
      },
    ],
  },
};
```

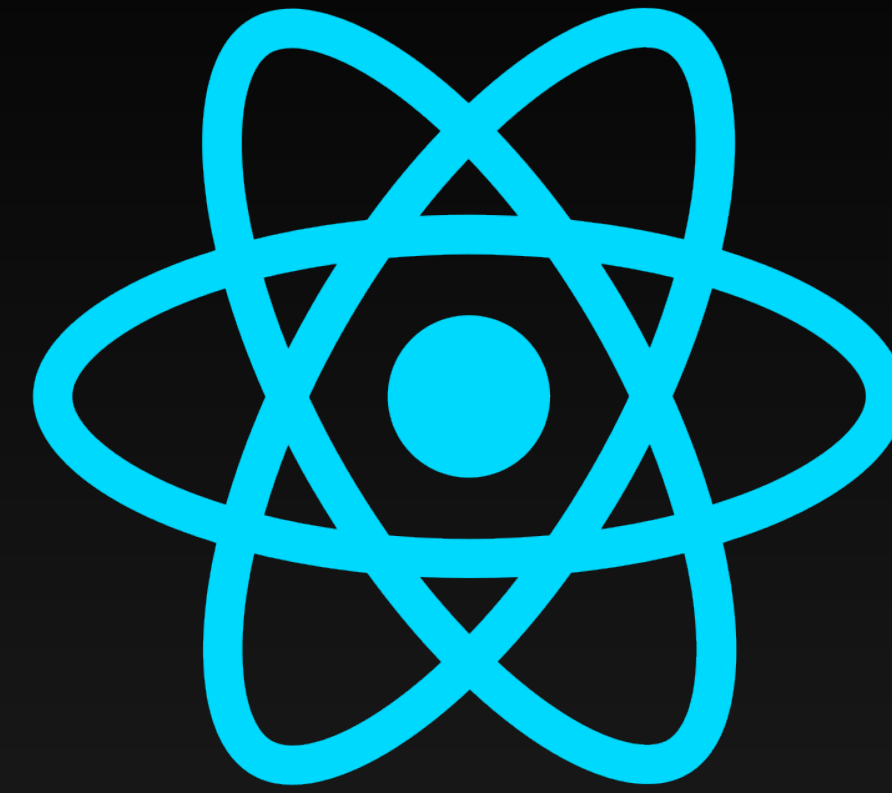
Additional Features

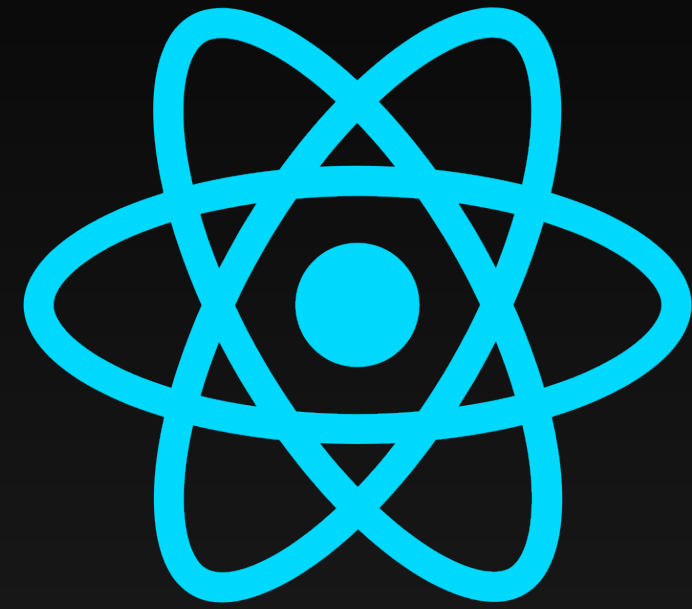
```
// What extra processing to perform
plugins: [
  new webpack.DefinePlugin({ ... }),
],

// Adjust module resolution algorithm
resolve: {
  alias: { ... },
},
```

Tool	Advantages	Gotchas	Ideal project
Browserify	Simplicity	Development is driven from community plugins only	Small project or prototype
Webpack	Flexible Configuration and Code Splitting	Config Readability	Medium-Large Web projects
Rollup.js	Bundling for Shareable Libraries	Modules bundling is already built into browsers	Library or Plugin
Gulp	Task Runner	Companion tool for a module bundler	Project in need of extra scripting
npm	Built in to most JavaScript projects	Manual set up	Works with most projects
Parcel	No config needed	Less documentation available	Medium-Large Web projects
Microbundle	Small footprint	Less documentation available	Size conscious Library or Plugin

<https://dev.to/netlify/choosing-a-javascript-build-tool-to-config-or-not-config-2ia8>





Bootstrap

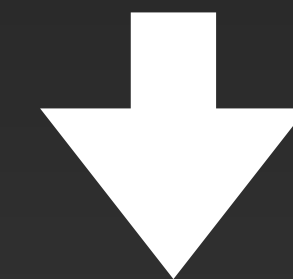
Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

[Get started](#)

[Download](#)

Currently v4.4.1



modules/bootstrap/...



- Repository for javascript packages
- Simple command line for adding packages to a project
- Provides a consistent language for libraries to make themselves available to others
- Tracks the list of dependencies for a module **along with the version**
- Provides a way to execute package binaries

```
{
  "name": "package",
  "version": "1.0.0",
  "description": "article",
  "main": "index.js",
  "author": "mrstalon",
  "license": "ISC",
  "dependencies": {
    "vue": "^2.5.13",
    "vue-material": "^1.0.0-beta-7",
    "vuex": "^3.0.1"
  }
}
```

